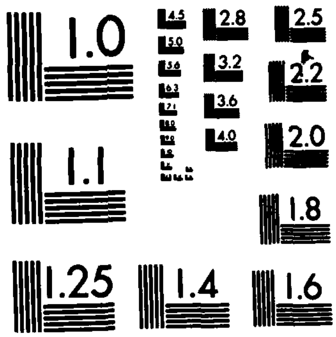
AD-A159 695 GENERATION OF VERTICALLY INCIDENT SEISMOGRAMS(U) HAWAII INST OF GEOPHYSICS HONOLULU L N FRAZER ET AL. SEP 85 HIG-CONTRIB-1643 N00014-82-C-0380 1/1

UNCLASSIFIED F/G 8/11 NL

END
FILMED
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A159 695

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

12

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER HIG Contrib. 1643	2. GOVT ACCESSION NO. AD-A159 695	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Generation of vertically incident seismograms.		5. TYPE OF REPORT & PERIOD COVERED
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) L. N. Frazer, D. L. Bates, and A. J. Rudman		8. CONTRACT OR GRANT NUMBER(s) N00014-82-C-0380
9. PERFORMING ORGANIZATION NAME AND ADDRESS Hawaii Institute of Geophysics 2525 Correa Road Honolulu, Hawaii 96822		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Ocean Sciences & Technology Division Bay St. Louis, MS 39520		12. REPORT DATE September, 1985 13. NUMBER OF PAGES 62 Pages
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Office of Naval Research, Branch Office 1030 East Green Street Pasadena, CA 91106		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved fo public release; distribution unlimited.

DTIC ELECTE OCT 8 1985 S D

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

A

18. SUPPLEMENTARY NOTES

Published in Geological Survey Occ. Pap. 49, Geophysical Computer Program 10, Dept. of Nat'l Resources, State of Indiana, Bloomington, Indiana, 62 pp., 1985

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Algorithm
Seismograms
FORTRAN 77
Vertically incident waves

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An algorithm for synthetic seismograms following the method of Kennett (1981) has been implemented in FORTRAN 77. Computations proceed in the frequency domain and then are Fourier inverted. Reflections of normally incident waves and their multiples are plotted on a Versatec plotter. The algorithm allows easy adjustment to other

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

plotting protocols. A brief outline of the theory and a program listing with examples are included.

Generation of Vertically Incident Seismograms

By L. NEIL FRAZER, DAVID L. BATES, *and* ALBERT J. RUDMAN

DEPARTMENT OF NATURAL RESOURCES
GEOLOGICAL SURVEY OCCASIONAL PAPER 49

SCIENTIFIC AND TECHNICAL STAFF OF THE GEOLOGICAL SURVEY

AUTHORS OF THIS REPORT: L. Neil Frazer and David L. Bates, Hawaii Institute of Geophysics, University of Hawaii at Manoa, 2525 Correa Road, Honolulu, HI 96822; Albert J. Rudman, Department of Geology, Indiana University, Bloomington, IN 47405.

Generation of Vertically Incident Seismograms

By L. NEIL FRAZER, DAVID L. BATES, *and* ALBERT J. RUDMAN

GEOPHYSICAL COMPUTER PROGRAM 10

DEPARTMENT OF NATURAL RESOURCES
GEOLOGICAL SURVEY OCCASIONAL PAPER 49

PRINTED BY AUTHORITY OF THE STATE OF INDIANA
BLOOMINGTON, INDIANA: 1985

STATE OF INDIANA
Robert D. Orr, *Governor*
DEPARTMENT OF NATURAL RESOURCES
James M. Ridenour, *Director*
GEOLOGICAL SURVEY
John B. Patton, *State Geologist*

For sale by Publications Section, Geological Survey, 611 North Walnut Grove,
Bloomington, IN 47405
Price $2.50

To the Geophysics Community

This report is one of a series of Geophysical Computer Programs that are being published in the Indiana Geological Survey Occasional Paper Series. Members of the Geophysics Section of the Indiana Geological Survey, with the advice and counsel of an advisory board,* select and edit submitted papers. Readers are invited to submit programs and manuscripts to the Geophysics Section. The primary purpose of this series is to make readily available those programs that deal with established geophysical computations.

Although the editors of some journals solicit only new approaches, we seek to publish programs that also deal with standard and classic problems. Our experience has shown that geophysicists, working alone or at relatively small laboratories, do not always have access to such programs. We also solicit programs implementing new geophysical procedures, but we anticipate that such material will be made available only rarely. Nevertheless, even large laboratories with extensive computer libraries may welcome a study of the other fellow's approach. In the same spirit, we hope that geophysicists will share both their new and standard programs.

The format for this series is intentionally kept simple to encourage others to submit manuscripts. It should contain: (1) a statement to establish the purpose of the program and some discussion of applications; (2) a brief summary of the theory that underlies the algorithm; (3) a discussion of the programs, perhaps with the aid of a flow diagram; and (4) presentation of a test case.

Responsibility for distribution of the program cards or furnished tapes will be assumed by the Indiana Geological Survey.

—Albert J. Rudman and Robert F. Blakely, editors

*Norman S. Neidell, Zenith Exploration Co., Inc.; Sigmund Hammer, University of Wisconsin; Judson Mead, Indiana University; Franklin P. Prosser, Indiana University; and Joseph E. Robinson, Syracuse University.

INDIANA GEOLOGICAL SURVEY GEOPHYSICAL COMPUTER PROGRAMS

No. 1 "Fortran Program for the Upward and Downward Continuation and Derivatives of Potential Fields" (Occasional Paper 10)

No. 2 "Fortran Program for Generation of Synthetic Seismograms" (Occasional Paper 13)

No. 3 "Fortran Program for Correlation of Stratigraphic Time Series" (Occasional Paper 14)

No. 4 "Fortran Program for Generation of Earth Tide Gravity Values" (Occasional Paper 22)

No. 5 "Fortran Program for Reduction of Gravimeter Observations to Bouguer Anomaly" (Occasional Paper 23)

No. 6 "Fortran Program for Correlation of Stratigraphic Time Series. Part 2. Power Spectral Analysis" (Occasional Paper 26)

No. 7 "Application of Finite-Element Analysis to Terrestrial Heat Flow" (Occasional Paper 29)

No. 8 "Generation of Synthetic Seismograms for an Acoustic Layer over an Acoustic Half Space" (Occasional Paper 35)

No. 9 "Computer Calculation of Two-Dimensional Gravity Fields" (Occasional Paper 40)

No. 10 "Generation of Vertically Incident Seismograms" (Occasional Paper 49)

Cost of Nos. 1 through 5 is $1.00 each + 50-cent mailing fee (third and fourth class) or 80-cent mailing fee (first class).

Cost of Nos. 6, 7, and 9 is $1.50 each + 50-cent mailing fee (third and fourth class) or 80-cent mailing fee (first class).

Cost of No. 8 is $2.00 + 50-cent mailing fee (third and fourth class) or 80-cent mailing fee (first class).

Cost of No. 10 is $2.50 + 50-cent mailing fee (third and fourth class) or 80-cent mailing fee (first class).

Contents

Illustrations

Illustrations

Tables

Generation of Vertically Incident Seismograms

By L. NEIL FRAZER, DAVID L. BATES, *and* ALBERT J. RUDMAN

Abstract

An algorithm for synthetic seismograms following the method of Kennett (1981) has been implemented in FORTRAN 77. Computations proceed in the frequency domain and then are Fourier inverted. Reflections of normally incident waves and their multiples are plotted on a Versatec plotter. The algorithm allows easy adjustment to other plotting protocols. A brief outline of the theory and a program listing with examples are included.

Introduction

This report concerns the generation of synthetic seismograms for vertically incident waves on a multilayered horizontal medium. Superposition of plane waves allows the simulation of a localized source. Computations proceed in the frequency domain to permit inclusion of attenuation by a complex velocity. The plane-wave components are related to reflection and transmission within the layers by a standard filter theory approach following the method of Kennett (1981). An expansion of the response into a power series permits the user to choose a complete-response seismogram or to designate the number of multiples to be used. The final seismogram is constructed by Fourier inversion.

The algorithm presented here, Program VISP, is modified from Kennett for the case of horizontally layered media bounded on top and bottom by half spaces. This code does not include a free surface, and therefore the ringing effect of surface multiples is not included.

Theory

In wave propagation in the presence of absorption, solution to the wave equation leads to a complex velocity V(w) that depends on the Q of the medium. The absorption model is an exponential decay with distance

$$A = A_o e^{-az} \tag{1}$$

where a is related to frequency w and Q

$$a \approx \frac{|w|}{2cQ} \tag{2}$$

and c is the phase velocity.

A plane compressional wave normally incident on two semi-infinite layers generates a reflected and transmitted wave. Continuity of displacement and stress determines the amplitudes of the waves in terms of the reflection and transmission coefficients R and T

$$R = \frac{p_j v_j - p_{j+1} v_{j+1}}{p_j v_j + p_{j+1} v_{j+1}} \qquad T = \frac{2 p_j v_j}{p_j v_j + p_{j+1} v_{j+1}} \tag{3}$$

where p_j is the density of layer j and v_j is the compressional velocity.

For n layers sandwiched between two half spaces, a matrix representation of the upgoing and downgoing wave fields U and D is given by

$$\begin{pmatrix} U_1 \\ D_1 \end{pmatrix} = Q_1 \; Q_2 \; \ldots \; Q_n \begin{pmatrix} U_{n+1} \\ D_{n+1} \end{pmatrix} \tag{4}$$

Q is a wave propagator related to a matrix of the total reflection and transmission coefficients R and T (Kennett, 1981, equation 3.81)

$$Q = \left(\begin{array}{c|c} T_U - R_D T_D{}^{-1} R_U & R_D T_D{}^{-1} \\ \hline -T_{DR_U} & T_{D}-1 \end{array}\right) \tag{5}$$

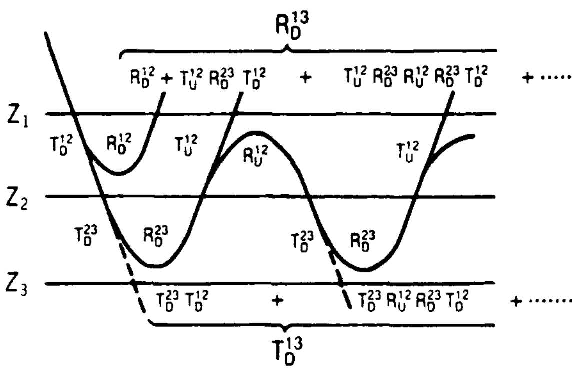


Figure 1. Graphic representation showing the first few terms of equation 9 for reflection and transmission matrices of superposed media and showing schematically the interactions undergone by the waves with the regions $z_1 < z < z_2$ and $z_3 < z < z_2$. From Kennett (1981).

where the subscripts U and D refer to computations for upgoing and downgoing waves.

To illustrate the significance of the total reflection and transmission coefficient, first consider two layers sandwiched between half spaces (fig. 1). Then the overall response in terms of reflection and transmission properties is

$$\begin{aligned}
R_D{}^{13} &= \overline{R}_D{}^{12} + \overline{T}_U{}^{12}\overline{R}_D{}^{23} \left[I - \overline{R}_U{}^{12}\overline{R}_D{}^{23}\right]^{-1} \overline{T}_D{}^{12} \\
T_D{}^{13} &= \overline{T}_D{}^{23} \left[I - \overline{R}_U{}^{12}\overline{R}_D{}^{23}\right]^{-1} \overline{T}_D{}^{12} \\
R_U{}^{13} &= \overline{R}_U{}^{23} + \overline{T}_{23}\overline{R}_U{}^{12} \left[I - \overline{R}_D{}^{23}\overline{R}_U{}^{12}\right]^{-1} \overline{T}_U{}^{23} \\
T_U{}^{13} &= \overline{T}_U{}^{12} \left[I - \overline{R}_D{}^{23}\overline{R}_U{}^{12}\right]^{-1} \overline{T}_U{}^{23}
\end{aligned} \tag{6}$$

```
      COMMON / CHAR / ITIME,IDATE,IOBT,TITLE
      WRITE(6,'(1X,"VISP OCEAN CRUST MODELING PROGRAM",8X,
     +               A10,2X,A10///)') ITIME,IDATE
      WRITE(6,'(" EPS = ",F6.4,"  SIGMA =",F5.2,3X,"YSIDE=",F5.2,
     +3X,"TLNTH = ",F5.2,/)')EPS,SIGMA,YSIDE,TLNTH
      WRITE(6,'(" LAYER#",1X,"VELOCITY",4X,"Q",5X,"DENSITY",1X,
     +"THICKNESS",7X,"DEPTH",/,9X,"KM/SEC",2X,"KM/SEC",3X,"GRM/SEC",
     +5X,"M",13X,"M")')
      WRITE(6,'("    1 ",F8.2,F9.1,F8.2)') CL(1),QL(1),RHO(1)
      TTOT = 0.0
      DO 40  I=2,NL-1
        WRITE(6,'(1X,I4,1X,F8.2,F9.1,F8.2,1X,F9.2,F10.2,
     + " TO",1X,F7.2)')I,CL(I),QL(I),RHO(I),T(I),TTOT,TTOT+T(I)
        TTOT = TTOT+T(I)
   40 CONTINUE
      WRITE(6,'(1X,I4,1X,F8.2,F9.1,F8.2,F20.2,
     +     " TO INFINITY")') NL,CL(NL),QL(NL),RHO(NL),TTOT
      RETURN
      END


C------------------------------------------------------------
C
C                SUBROUTINE PLTARMI
C
C------------------------------------------------------------

      SUBROUTINE PLTARMI
      COMMON / NUM / CL(200),QL(200),RHO(200),T(200),EPS,SIGMA,
     +               CRUST,NL,NTRACE,LOBS,NW,TSEC,TSC,ASC,NMULT,
     +               TLAG,YSIDE,TLNTH
      COMMON / CHAR / ITIME,IDATE,IOBT,TITLE
      COMMON /FILTDT/ F1(10),IDB(10),IFILTYP(10),NFILT,MPHASE,
     + FF1(10)
      COMMON  /AR/IAGC,WINDOW
      LOGICAL MPHASE
      CHARACTER*80 TITLE
      CHARACTER*10 ITIME,IDATE,LINE
      CHARACTER*1 IOBT
      NFILT = 0
      READ(1,'(A2)') WWW
      READ(1,*) NTRACE,NW,TSEC,TSC,ASC,NMULT,IAGC,
     *WINDOW
      WRITE(6,101)
  101 FORMAT(//,1X,"NTRACE",1X,"# FREQS",1X,"T(MSECS)",1X,"IN/MSECS",
     +1X,"AMP-SCL",1X,"MULTPL",1X,"AGC",1X,"WINDOW")
      WRITE(6,102)NTRACE,NW,TSEC,TSC,ASC,NMULT,IAGC,WINDOW
  102 FORMAT(3X,I2,4X,I4,3X,F6.0,4X,F4.3,6X,F4.1,4X,I2,3X,
     +I3,2X,F4.0)
        IF (NTRACE .GT. 10.0) CALL TERMIN(27,0)
      IF ((NW.NE.16) .AND. (NW.NE.32) .AND. (NW.NE.64) .AND. (NW.NE.128)
     +    .AND. (NW.NE.256) .AND. (NW .NE. 512)) CALL TERMIN(23,NW)
      IF (TSEC .LT. 1.0) CALL TERMIN(25,NINT(TSEC))
      IF ((TSC.EQ.0.0) .OR. (TSC*TSEC.GT.YSIDE)) THEN
        TSC = YSIDE/TSEC
```

```
      IF (NINTLAY+I .GT. NL) CALL TERMIN(4,MLNR)
      TINTLAY = RHO(I)
      IF (TINTLAY .LT. 0) THEN
        TINTLAY = -TINTLAY-CRUST
        IF (TINTLAY .LT. 0) CALL TERMIN(5,MLNR)
      ENDIF
      CRUST = CRUST+TINTLAY
      INCRCL = (CL(I+NINTLAY)-CL(I-1))/(NINTLAY+1)
      INCRQL = (QL(I+NINTLAY)-QL(I-1))/(NINTLAY+1)
      IF (RHO(I+NINTLAY).EQ.-1) RHO(I+NINTLAY)=(CL(I+NINTLAY)+1.5)/3
      INCRRHO = (RHO(I+NINTLAY)-RHO(I-1))/(NINTLAY+1)
      DO 25 J=1,NINTLAY
        CL(J+I-1) = CL(I-1) + J*INCRCL
        QL(J+I-1) = QL(I-1) + J*INCRQL
        RHO(J+I-1) = RHO(I-1) + J*INCRRHO
        T(J+I-1) = TINTLAY/NINTLAY
25    CONTINUE
      I = I+NINTLAY
    ELSE
      IF (RHO(I) .EQ. -1) RHO(I)=(CL(I)+1.5)/3
      IF (T(I) .LT. 0) THEN
        T(I) = -T(I)-CRUST
        IF (T(I) .LT. 0) CALL TERMIN(5,MLNR)
      ENDIF
      IF (I .LT. NL) THEN
        CRUST = CRUST+T(I)
      ELSE
        T(I) = 10000
      ENDIF
      I = I+1
    ENDIF
    IF(I.LT.NL)GO TO 20
    IF(RHO(I).EQ.-1) RHO(I) =(CL(I)+1.5)/3
    DO 30 I=2,NL
      IF ((CL(I).LE.0) .OR. (CL(I).GE.10)) CALL TERMIN(11,I)
      IF (QL(I) .LT. 0) CALL TERMIN(12,I)
      IF(((RHO(I).LE.0).AND.(RHO(I).NE.-1)).OR.(RHO(I).GE.10))
   *  CALL TERMIN(15,I)
30  CONTINUE
    END


C-----------------------------------------------------------
C
C           SUBROUTINE PRTMLDA
C
C-----------------------------------------------------------

    SUBROUTINE PRTMLDA
    CHARACTER*80 TITLE
    CHARACTER*10 ITIME,IDATE
    CHARACTER*1 IOBT
    COMMON / NUM / CL(200),QL(200),RHO(200),T(200),EPS,SIGMA,
   +               CRUST,NL,NTRACE,LOBS,NW,TSEC,TSC,ASC,NMULT,
   +               TLAG,YSIDE,TLNTH
```

```
      END

C-----------------------------------------------------------
C
C              SUBROUTINE MDLINCH
C-----------------------------------------------------------

      SUBROUTINE MDLINCH
      CHARACTER*80 TITLE
      CHARACTER*10 ITIME,IDATE
      CHARACTER*1 IOBT
      COMMON / NUM / CL(200),QL(200),RHO(200),T(200),EPS,SIGMA,
     +                CRUST,NL,NTRACE,LOBS,NW,TSEC,TSC,ASC,NMULT,
     +                TLAG,YSIDE,TLNTH
      COMMON / CHAR / ITIME,IDATE,IOBT,TITLE
      REAL          INCRCL,INCRQL,INCRRHO,TINTLAY
      TINTLAY=0.0
      ILNR=0
C
C     READ IN TITLE AND WRITE IT
C
      READ(1,100)TITLE
  100 FORMAT(A)
      WRITE(6,200)TITLE
  200 FORMAT(///,1X,A)
      NL=-1
      ILNR = ILNR+1
      DO 10 I=1,200
        NL = NL+1
        IF (NL .EQ. 0) THEN
          READ(1,'(A2)')WWW
          READ(1,*) EPS,SIGMA,YSIDE,TLNTH
          READ(1,'(A2)')WWW
        ELSE
          READ(1,*) CL(NL),QL(NL),RHO(NL),T(NL)
          IF(CL(NL).EQ.9999)GO TO 15
C
C         ...INTERPOLATED LAYERS
C
          IF (CL(NL) .EQ. 0) NL=NL+NINT(QL(NL))-1
          IF (NL .GT. 200) CALL TERMIN(1,ILNR)
        ENDIF
   10 CONTINUE
 15   IF ((CL(1) .EQ. 0) .OR. (T(1)  .LT. 0)) CALL TERMIN(3,1)
      NL = NL-1
      CRUST = 0
      MLNR=1
      I=2
C
C      ...EXPAND INTERPOLATED LAYERS AND CONVERT ABSOLUTE DEPTHS TO THICKNESS.
C
 20   MLNR = MLNR+1
      IF (CL(I) .EQ. 0) THEN
        NINTLAY = NINT(QL(I))
```

```
C ----------------
C
C
C  THE CODE HAS A MAIN PROGRAM THAT CALLS 6 SUBROUTINES. THESE
C  SUBROUTINES, IN TURN, CALL 6 OTHER SUBROUTINES. OUTPUT CONSISTS
C  OF A BRIEF LINE PRINTER SUMMARY OF THE INPUT PARAMETERS AND A
C  PLOT OF THE SYNTHETIC SEISMOGRAM AND LAYER PARAMETERS (VELOCITY,
C  Q AND DENSITY).
C
C ACTION OF PROGRAM:
C ----------------

C   THE PROGRAM READS AND CHECKS THE MODEL PARAMETERS.  IN ALL CASES IT THEN
C   PLOTS THE PARAMETER VALUES AS THEY VARY WITH DEPTH.  THEY ARE PLOTTED ON
C   THE SAME GRAPH, WITH VALUES OF Q REDUCED BY 1000.  THE Y AXIS FILLS THE
C   PAPER WIDTH AND HAS VALUES 0 - MAX PARAMETER VALUE.  THE LENGTH OF THE X
C   AXIS IS DEPENDENT ON BOTH THE TOTAL THICKNESS AND THE NUMBER OF LAYERS
C   REQUESTED IN THE MODEL.
C
C   THE MODEL USES THE THEORY OF B.L.N.KENNETT (ADVANCES IN APPLIED
C   MECHANICS, VOLUME 21, PP.79-167, ACADEMIC PRESS, 1981) TO COMPUTE
C   RESPONSE OF MODEL FOR  SELECTED FREQUENCIES (SUBROUTINE EXEMODL)
C   2*PI/TSEC,....,NW*2*PI/TSEC.   THE RESPONSE VALUES ARE THEN
C   FOURIER TRANSFORMED TO THE TIME DOMAIN AND PLOTTED.
C   IF NO TIME SCALE FACTOR IS SPECIFIED (OR IF THE ONE GIVEN IS
C   INAPPROPRIATE), A VALUE OF 1 MSEC PER INCH OR LESS, IF NECESSARY, IS USED.
C   TRACES ARE SPACED 0.65 INCHES APART AND ARE SCALED TO A MAXIMUM MAGNITUDE
C   OF 0.5 INCHES, UNLESS AN ASC VALUE IS DESIGNATED (OTHER THAN 1). IN THAT
C   CASE, PEAKS ARE TRUNCATED AT 0.5 INCHES.
C
C---------------------------------------------------------------------------
C---------------------------------------------------------------------------

C
C                        MAIN PROGRAM
C
C--------------------------------------------------------------------------------
      PROGRAM TEST(OUTPUT,TAPE6=OUTPUT)
      CHARACTER*80 TITLE
      CHARACTER*10 DATE,TIME,ITIME,IDATE
      CHARACTER*1 IOBT
      COMMON / NUM / CL(200),QL(200),RHO(200),T(200),EPS,SIGMA,
     +              CRUST,NL,NTRACE,LOBS,NW,TSEC,TSC,ASC,NMULT,
     +              TLAG,YSIDE,TLNTH
      COMMON / CHAR /ITIME,IDATE,IOBT,TITLE
      COMMON   /AR/IAGC,WINDOW
      OPEN(UNIT=1,STATUS='OLD',FILE='MDL')
      ITIME=TIME()
      IDATE=DATE()
      CALL MDLINCH
      CALL PRTMLDA
      CALL PLTARMI
      CALL EXEMODL
      IF(IAGC.EQ.1)CALL AGCFREQ
      CALL WRTPLT
```

```
C          AMPLIFY ALL EVENTS AND CREATE A RINGING EFFECT. TOO LARGE
C          A WINDOW EFFECTIVELY CANCELS THE AGCACTION.
C
C       8) CAPTIONS (NOT READ BY THE COMPUTER)
C
C       9) PARAMETERS SELECTING DETECTOR AND LAYER TO PUT IT
C                 LINE,LOBS
C
C          WHERE
C
C          LINE - IS EITHER "REFLECTION", "VERTICAL" OR "PRESSURE"
C          DEPENDING ON RECEIVER TYPE DESIRED
C          LOBS - IS LAYER NUMBER TO PLACE RECEIVER (NUMBER MUST
C          BE IN COL 28)
C
C          IF THE RECEIVER CHOICE IS THE REFLECTION COEFFICIENT
C          ONLY LAYER 1 MAY BE DESIGNATED. THE CALCULATION IS
C          MADE AT THE BASE OF THE HALF SPACE. IF THE RECEIVER IS
C          DESIGNATED AS VERTICAL(DISPLACEMENT) OR PRESSURE, THE
C          RECEIVER IS AN OBS OR BOREHOLE INSTRUMENT AND MUST BE
C          PLACED IN LAYER 2 OR GREATER (NOT LAYER 1). THE OBS
C          IS AT THE TOP OF THE LAYER SPECIFIED,BUT WITHIN THE
C          LAYER. FOR EXAMPLE, AN OBS IN LAYER 2 IS AT THE TOP
C          OF LAYER 2, ESSENTIALLY ACROSS THE INTERFACE FROM THE
C          STANDARD REFLECTION COEFFICIENT CALCULATION IN
C          LAYER 1.
C
C      10) CAPTIONS (NOT READ BY COMPUTER)
C
C      11) IF THE PLOT IS IS TO BE FILTERED, THE FILTER PHASE IS GIVEN
C                   IPHASE
C
C          IPHASE - IF "0" THE FOLLOWING FILTER(S) IS ZERO PHASE,
C                   IF "1" THE FILTER(S) ARE MINIMUM PHASE
C
C      12) SUBSEQUENT DATA LINES (UP TO 10) ARE FILTER DEFINITIONS
C                   IFILTYP,IDB,FF1
C
C          WHERE:
C
C          IFILTYP - 1 = HIGH CUT (LOW PASS) BUTTERWORTH
C                    2 = LOW CUT (HIGH PASS) BUTTERWORTH
C                    3 = NOTCH FILTER
C          IDB - DECIBEL REDUCTION WHERE POLES = DB/6 + 1
C          FF1 - CUT-OFF FREQUENCY IN HZ
C
C          IF A BAND PASS FILTER IS DESIRED, TWO DATA LINES ARE
C          READ IN CONSECUTIVELY: ONE GIVING THE HIGH CUT VALUES
C          THE OTHER THE LOW-CUT.
C
C          IF A NOTCH FILTER IS DESIRED, INPUT IS SIMILAR TO
C          BAND PASS (SEE ABOVE), EXCEPT IFILTYP = 3 FOR
C          BOTH INPUT LINES
C
C  USING THE PROGRAM:
```

```
C          TO ZERO. THE FINAL LAYER IS THE BOTTOM HALF-SPACE AND IS IDENTIFIED
C          BY A THICKNESS T SET EQUAL TO ZERO.  A FLAG OF 4 CONSECUTIVE 9999'S
C          CLOSES OFF THE LAYER PARAMETER INPUT.
C
C          FOR EACH LAYER, OTHER THAN THE FIRST,IF RHO IS GIVEN AS A -1, A
C          DEFAULT VALUE OF (CL+1.5)/3 IS TAKEN.
C
C          EXTRAPOLATION - IT MAY BE DESIRED TO HAVE LAYER PARAMETERS CHANGE
C          SMOOTHLY FOR A CERTAIN REGION.  TO AVOID HAVING TO HAND CALCULATE AND
C          INPUT VALUES FOR SUCH A REGION DIVIDED INTO A DISCRETE NUMBER OF
C          LAYERS, IT IS POSSIBLE TO REQUEST THAT BETWEEN THE PRIOR AND NEXT
C          FULLY DESCRIBED LAYERS THERE WILL BE N LAYERS OF EQUAL THICKNESS AND
C          WITH PARAMETERS COMPUTED BY LINEAR EXTRAPOLATION BETWEEN THE VALUES
C          FOR THE PRIOR AND NEXT LAYERS.  THIS OPTION IS REQUESTED BY ENTERING
C          A LINE OF THE FORMAT 0,N,T WHERE THE 0 IS AN ESCAPE VALUE FOR THE
C          NORMALLY EXPECTED CL, N IS THE NOS OF LAYERS TO BE INSERTED, T
C          IS THE TOTAL THICKNESS OF THOSE LAYERS, AND THE "1" IS A DUMMY.
C          FOR EXAMPLE,THE VALUE OF CL FOR THE 2ND INTERPOLATED LAYER WILL BE:
C                    CL(PRIOR) + 2*(CL(NEXT) - CL(PRIOR))/(N+1)
C          AS A USEFUL GUIDE, IF YOU WANT EACH LAYER 1/4 WAVELENGTH THICK (AT
C          THE HIGHEST FREQUENCY TO BE USED), CHOOSE   N = T*NW*4/(CL*TSEC)
C
C          ABSOLUTE DEPTH - NORMALLY T SHOULD BE A POSITIVE VALUE
C          SPECIFYING THE THICKNESS OF THE LAYER.  IF, HOWEVER, IT IS PREFERRED
C          TO SPECIFY THE DEPTH BELOW THE HALF SPACE TO WHICH THE LAYER WILL
C          EXTEND (ITS STARTING POINT WILL OF COURSE BE IMMEDIATELY BELOW THE
C          PREVIOUSLY DESCRIBED LAYER), A NEGATIVE VALUE CAN BE GIVEN, WHOSE
C          MAGNITUDE WILL INDICATE THAT ABSOLUTE DEPTH.  THIS OPTION CAN ALSO BE
C          BE USED WITH EXTRAPOLATED LAYERS.
C
C        6) CAPTIONS (NOT READ BY COMPUTER)
C
C
C        7) INPUT PARAMETERS FOR COMPUTATIONAL PURPOSES IN FREE FORMAT ARE:
C               NTRACE,NW,TSEC,TSC,ASC,NMULT,IAGC,WINDOW
C
C           WHERE:
C
C           NTRACE -NUMBER OF TRACES PLOTTED AS DUPLICATES ON THE SYNTHETIC
C           NW - NOS OF FREQUENCIES (ł16, POWER OF 2) TO BE USED IN FOURIER
C           TRANSFORM TO TIME DOMAIN.
C           TSEC - MSECS OF TIME SERIES (STARTS AT -1, -5 OR -10 MSECS
C           DEPENDING ON THE VALUE OF TSEC).  TSEC SHOULD BE MORE
C           THAN TIME FOR 1-3 REFLECTIONS FROM BOTTOM INTERFACE, DEPENDING
C           ON WHETHER YOU ARE REQUESTING PRIMARIES ONLY OR FULL RESPONSE.
C           TSC - TIME SCALING FACTOR(IN/MSEC).IF TSEC*TSCłYSIDE,THEN IF
C           TSC IS POSITIVE,A VALUE OF TSC IS COMPUTED = YSIDE/TSEC, WHILE
C           IF TSC IS NEGATIVE,PLOT IS TRUNCATED AT YSIDE/TSC MILLISECONDS
C           ASC- AMPL SCALE FACTOR (DEFAULT = ł1.0).  IF ł 1, SIGNAL IS
C           AMPLIFIED, BUT THE PLOT IS CLIPPED AT 0.5 INCHES AMPLITUDE.
C           NMULT -NOS OF MULTIPLE REFLECTIONS TO BE COMPUTED.
C           VALUE ł 0 IMPLIES FULL RESPONSE REQUIRED (BY MATRIX INVERSION).
C           IAGC- SET = TO 1 IF AUTOMATICGAIN CONTROL IS DESIRED
C           WINDOW - LENGTH OF AGCWINDOW IN MSECS. A SMALL WINDOW
C           (RELATIVE TO EXPECTED TIME INTERVALS OF REFLECTIONS) WILL
```

Appendix 3. FORTRAN 77. Program VISP and Program PLTVISP

```
C                 ======================================
C
C                              PROGRAM VISP
C
C                 (VERTICAL INCIDENT SEISMOGRAM PROGRAM)
C
C           BY L.N. FRAZER, D.L. BATES & A.J. RUDMAN JULY 1983
C
C
C
C  USES I/O STREAMS AS FOLLOWS:
C  ---------------------------
C    1:  INPUT
C    6:  OUTPUT, INCLUDING ERROR MESSAGES
C    4:  STORED ON FILE PLTIN FOR PLOTTING BY PROGRAM PLTVISP
C
C
C  FORMATS AND DESCRIPTIONS OF INPUT FILE:
C  --------------------------------
C   A) DATATYPES - ALL DATA IS ACCEPTED IN FREE FORMAT, WITH ANY EMPTY FIELDS
C         BEING READ AS 0.
C
C   B)  DATALINES - USEABLE DATALINES ARE INTERSPERSED WITH CAPTIONS-LINES
C
C       1) TITLE (ALPHA-NUMERIC)
C
C       2) CAPTIONS (NOT READ BY COMPUTER)
C
C       3) EPS,SIGMA,YSIDE,TLNTH
C
C            WHERE:
C
C            EPS - PARAMETER GOVERNS VARIATION OF VELOCITY WITH FREQUENCY W
C            SIGMA - PARAMETER GOVERNS VARIATION OF VELOCITY WITH FREQUENCY W
C            YSIDE -  MAXIMUM INCHES PERMITTED FOR LENGTH OF SYNTHETIC PLOT
C            TLNTH -  FIXED LENGTH OF PARAMETER PLOT(IN INCHES)
C
C       4) CAPTIONS (NOT READ BY COMPUTER)
C
C       5) CL,QL,RHO,T
C
C            WHERE:
C
C            CL  - IS COMPRESSIONAL VELOCITY OF LAYER IN KM/SEC
C            QL  - IS COMPRESSIONAL Q FACTOR
C            RHO - IS DENSITY IN GM/CC
C            T   - IS THICKNESS IN METERS
C
C         ALL SUBSEQUENT DATA LINES ARE READ AS DESCRIPTIONS OF CONSECUTIVE
C         LAYERS OF THE MODEL (UP TO 200). THE FIRST LAYER IS THE UPPER
C         HALF-SPACE (CORRESPONDING TO AN OCEAN LAYER) WITH THICKNESS SET
```

RVRB2EW	RVRB2*EWIGH*TU
SC	FFT variable
SIGMA	Parameter used in computing variation of velocities with W
SIGNI	FFT variable
SQREFT	Square of amplitude
SQREFW	Frequence equivalent of SQREFT
T	Array of thickness (in meters), one for each layer
TD	Downward transmission coefficients for current interface
TDON	Downward transmission coefficients from ocean bottom to lower half space
TDONP	New value of TDON
TDOR	Downward transmission coefficients from ocean bottom to receiver
TEMP	Temporary variable for truncation
TEST	Temporary variable
TINTLAY	Thickness of each interpolated layer
TITLE	String used for plotting
TLAG	Time plots will begin at -TLAG msec
TLNTH	Fixed length of parameter plot (in inches)
TSC	Time-scaling factor in inches/msec
TSEC	Number of msec of time series
TTOT	Total thickness (in meters)
TU	Upward transmission matrix for current interface
TUNO	Upwards transmission matrix from lower half space
TUNOP	New value of TUNO
UNIT	CMPLX (1.0, 0.0)
W	Current frequency in KHz
WARNED	Logical variable
WIGH	Phase lag associated with propagation through vertical distance
WINDOW	AGC window length in msec
WW	AGC filter
WWW	Ruse to read past a card
Y	Common factor used in routine REFL
YSIDE	Max length of plot (in inches)

ISTEP	FFT variable
ITIME	Current time
ITW2	Number of points in AGC filter
IW	Frequency loop counter
J	Miscellaneous loop counter
L	FFT variable
LINE	String to hold lines read from input file (allows free format). (See IOBT.)
LOBS	Layer in which OBS located (1 if no OBS)
LX	FFT variable
M	FFT variable
MAXABSV	Filter variable
MD	(See Kennett, 1981, p. 106, equation 3.38.)
MPHASE	Flag: If set, Minimum-phase filter required; otherwise, Zero phase
MLNR	Counter of model file input lines, used in error messages
MU	(See Kennett, 1981, p. 106, equation 3.38.)
NINTLAY	Number of interpolated layers requested
NFILT	Number of filters requested
NL	Total number of layers in model including bounding half spaces
NMULT	Number of multiple reflections to be calculated
NT	Number of time steps (= NW*2)
NTRACE	Number of traces to be plotted on the seismogram
NW	Number of frequencies to be calculated
PI	3.141592653
POLES	Filter variable
PR	Value of pressure recorded at OBS
PRESSFA	Pressure related variable for given layer
QL	Array of P-wave Q values, one for each layer
R	Filter variable
RCOEFF	Reflection coefficient or OBS response for current frequency W and slowness
RD	Downward reflection matrix for current interface
RD0N	Downward reflection matrix from ocean bottom to lower half space
RD0NP	New value of RD0N
REFT	Reflection coefficient (or OBS response) in time domain
REFW	Reflection coefficient (or OBS response) in frequency domain
RHO	Array of densities, one for each layer
RHO1	Density of layer above the current interface
RHO2	Density of layer below the current interface
RTB1	Temporary variable
RTB2	Temporary variable
RTB3	Variable in computation of RCOEFF
RU	Upward reflection coefficients for current interface
RUN0	Upward reflection coefficients from lower half space to ocean bottom
RUN0P	New value of RUN0
RUR0	Upward reflection coefficients from receiver to ocean bottom
RVRB1	Effect on downward waves of all multiples in current layer
RVRB1EW	RVRB1*EWIGH*TD0N
RVRB2	Effect on upward waves of all multiples in current layer

Appendix 2. Glossary of Variables Used in Program VISP

(Variables in the plotting Program PLTVISP are not listed here)

A	Dummy argument in multiple computation
AA	FFT variable
AL1	P-wave slowness of the layer above the current interface
AL2	P-wave slowness of the layer below the current interface
ASC	Amplitude scaling factor for traces
B	Dummy argument in multiple computation
BB	AGC filter variable
BEX	Numbers smaller than exp (-BEX) taken to be zero
C0	CMPLX (0.0.0.0)
C1	CMPLX (1.0.0.0)
CL	Array of P-wave velocities, one per layer
CRUST	Total thickness of layers, excluding first and last
CTEMP	FFT variable
CW	FFT variable
CX	FFT variable
CURRMIN	AGC filter variable
DELFREQ	Frequency increment in filtering
DELTAT	Time increment
DET	Temporary variable in multiple computation
EIW	CMPLX(EPS,-W)**SIGMA
EPS	Parameter governing variation of velocities with frequency W
EWIGH	EXP(WIGH)
EWRD	EWIGH*RD
EWRUN0	EWIGH*RUN0
F1	Cut off frequency in Khz
FF1	Cut off frequency in Hz
FACTOR	Filter variable
FILT	Digitized filter
FILT1	Second copy of filter
FRSTTIM	Logical variable
GL	Z-comp of P-slowness vector
GL1	Z-comp of P-slowness vector in layer above current interface
GL2	Z-comp of P-slowness vector in layer below current interface
I	Miscellaneous loop counter
IAGC	Flag for AGC filter
IDATE	Current date
IDB	DB reduction for notch filter, or =6*POLES-1 for Butterworth
IFILTYP	Type of each filter requested
IL	Layer loop counter
INCRCL	Incremental value of CL through interpolated layers
INCRQL	Incremental value of QL through interpolated layers
INCRRHO	Incremental value of RHO through interpolated layers
INDEX1	Filter variable
INDEX2	Filter variable
IOBT	Character for type of OBS response: reflection/vertical/pressure
IPHASE	Flag: 0 for zero phase and 1 for minimum phase

Appendix 1. Generalized Description Diagram of Programs VISP and PLTVISP

Program VISP

Subroutine	Comments
1. MAIN	Calls six subroutines listed below.
2. MDLINCH	Reads model parameters and interpolates layer parameters. Calls TERMIN for error stop.
3. PRTMLDA	Writes model parameters.
4. PLTARMI	Reads and writes computational parameters, user options, and filter specifications. Calls TERMIN for error stop.
5. EXEMODL	Main computation subroutine for a given frequency: Computes reflection coefficient for all layers (for user-designated number of multiples). Options for pressure or vertical displacement. Repeats for all frequencies and then filters. Finally, transforms to time domain.
	The above computations involve calls to five subroutines: PARMGEN (computes vertical slowness), REFL (computes reflection coefficient for a single interface at a single frequency), REVERB (selects multiple computations), FILTER (filters amplitudes), and FFT (fast Fourier transform).
6. AGCFREQ	Simulates automatic gain control in frequency domain for specified window. Calls FFT.
7. WRTPLT	Writes header and seismogram amplitudes on file for plotting by Program PLTVISP. (See table below.)

Program PLTVISP

Subroutine	Comments
1. MAIN	Calls four subroutines listed below.
2. READINP	Reads data generated by Program VISP.
3. INITPLT	Plots major headings.
4. PARMPLT	Initializes plot parameters and plots axes and model parameters.
5. RSPMDL	Writes filter, AGC, and multiple specifications. Plots seismogram traces and axes.

typical field records. As a demonstration, a model that simulates a continuous velocity log with 123 layers was used to create Test 1 (fig. 2). A CDC Cyber 170/855 computer at the Indiana University Wrubel Computing Center generated the output data in 13.6 seconds with a 71300 octal field length. Table 1 lists the input data used to generate figure 2.

Appendix 4 lists the input data and output plots for 12 other test cases used to demonstrate the user options.

Literature Cited

Kennett, B. L. N.

1981 - Elastic wave propogation in stratified media, *in* Advances in applied mechanics: New York, Academic Press, Inc., v. 21, p. 79-167.

Table 1. Input data used to generate a typical synthetic seismogram (fig. 2)

```
TEST 1 - DSYN.  SYNTHETIC SEISMOGRAM.
EPS      SIGMA    YSIDE     XSIDE
0.001     0.1      40.       12.0
CL        QL       RHO      THICKNESS
1.5      2000.     1.1          0.
1.9        22.     2.2         10.
2.1        22.     2.2         10
2.0        22.     2.2         10
2.2        22.     2.2         10
2.1        22.     2.2         10
2.8        25.     2.5         10
2.9        25.     2.5         10
 .          .       .           .
 .          .       .           .
 .          .       .           .
 .          .       .           .
2.7        23      2.4         10
2.6        23      2.4         10
2.7        23      2.4         10
2.8        23      2.4         10
2.6        23      2.4         10
1.5        20      2.2          0.
9999     9999     9999       9999
#TRC  #FREQ T(MSEC)   IN/MSEC  AMP-SCALE   MLTPL   AGC   WINDOW
 5     512  2000.       .006       1.0       -1      1     400
DETECTOR TYPE      DETECTOR LOCATION
 REFLECTION              LAYER 1
FILTER PHASE
1  (MINIMUM PHASE)
FILTER-TYPE   DB  SLOPE    CUT-OFF FREQUENCY
1                  72               70
2                  72               30
```

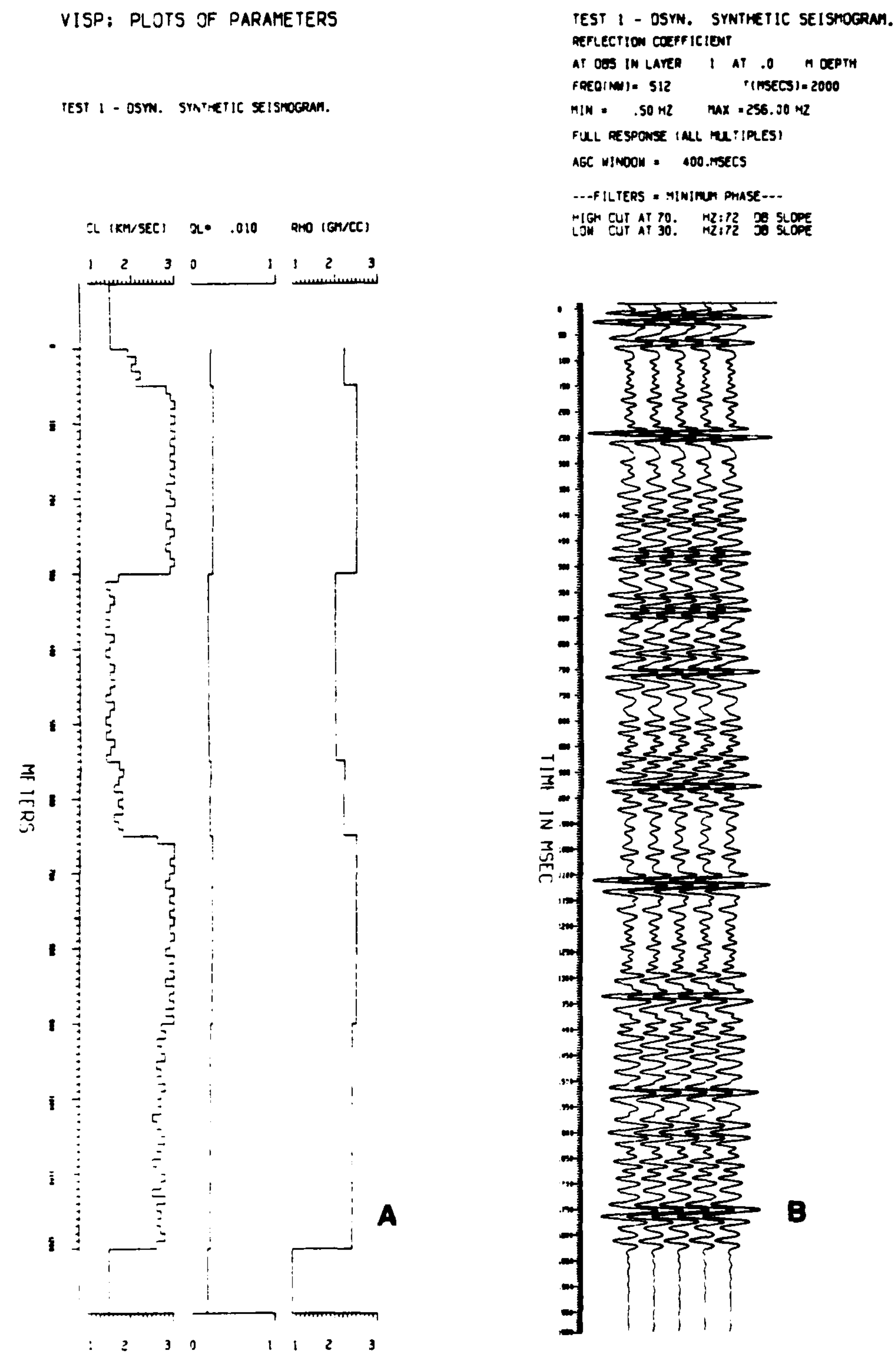


Figure 2. (Test 1) Output plots generated by Program VISP. A, input data: velocity (CL), Q (QL), and density (RHO); B, synthetic seismogram generated from input data. Significant parameters are listed above the seismogram.

where, for example, the coefficients $\overline{R}_D^{12}$ are for the interface at Z_2, and the coefficients $\overline{R}_D^{23}$ are for the interface at Z_3, but are adjusted for travel through the layer by $E = e^{i\omega d/c}$, where d is layer thickness. Hence

$$\overline{R}_D^{23} = ER_D^{23}E \quad \overline{R}_U^{23} = ER_U^{23}E \quad \overline{T}_D^{23} = ET_D^{23} \quad \overline{T}_U^{23} = ET_U^{23} \tag{7}$$

Kennett (1981) pointed out that the bracketed quantity may be expanded as a power series

$$[I - A]^{-1} = I + A + A^2 + \ldots \tag{8}$$

Therefore,

$$R_D^{13} = R_D^{12} + T_U^{12}R_D^{23}T_D^{12} + T_U^{12}R_D^{23}R_U^{12}R_D^{23}T_D^{12} + \ldots \tag{9}$$

which is represented schematically in figure 1. R_D^{12} is the reflection from the upper region. Reading from right to left, $T_U^{12}\ R_D^{23}\ T_D^{12}$ represents a transmission back up through the upper zone. $T_U^{12}\ R_D^{23}\ R_U^{12}\ R_D^{23}\ T_D^{12}$ represents a reverberation within the layer.

The total response (equation 6) includes all internal reverberations, and the bracketed quantity is termed the reverberation operator. Note that the reflection-transmission coefficient may be calculated iteratively by adding a layer at each stage, beginning at the top or the bottom of the stack of layers. Computation of the response of the model proceeds for selected frequencies and is finally Fourier transformed to the time domain and plotted.

Summary

The algorithm for Program VISP and Program PLTVISP (appendix 1) presumes an incident plane wave originating at the base of, but within, the upper half space. The two programs are written in FORTRAN 77, and the codes are listed in appendix 3. Only one input file is required to run Program VISP. One output file is generated for plotting. Program PLTVISP plots the output on a Versatec plotter by using standard calcomp calls. A glossary of constants and variables is in appendix 2. The comment cards at the beginning of Program VISP detail, card by card, the input data required to generate an output.

User's options include an output plot (seismogram) of the reflection coefficient (pressure of the reflected wave divided by the pressure of the incident wave), pressure, or vertical displacement. The receiver may be at the top of any of the layers except when a reflection coefficient is requested; then the receiver is just above the first interface. Seismograms may be filtered by using a Butterworth-type high-pass, low-pass, or band-pass filter. The filter slopes and the phase characteristics (minimum or zero) are specified in the input. The user may uniformly amplify the signal (ASC) or optionally select a simulated automatic gain control (AGC). The final seismogram may display primary reflections only, primaries and a selected number of multiples, or a complete response.

Besides computational parameters, the user must read in the following model parameters for each layer and the top and the bottom half space: velocity, Q, density, and thickness. If a negative one is read in for the density, a default value of (velocity + 1.5)/3 is calculated. Program VISP successfully generates synthetic seismograms that simulate

```
      ELSEIF (TSC.LT.0.0) THEN
C
C     ...IN THIS CASE DONT PLOT ALL.
C
        TSC = -TSC
      ENDIF
      IF (ASC .EQ. 0.0) ASC=1.0
      IF ((ASC .LT. 1.0) .OR. (ASC .GT. 100)) CALL TERMIN(29,0)
C
C       LOOK FOR OBS OPTIONS
C       -------------------
C
        READ(1,'(A2)')WWW
      READ(1,'(A11,16X,I1)')LINE,LOBS
        WRITE(6,103)LINE,LOBS
 103    FORMAT(/,1X,"DETECTOR TYPE IS ",A," IN LAYER #",I3)
        IOBT=LINE(1:1)
        IF ((IOBT.EQ.'V') .OR. (IOBT.EQ.'P')) THEN
          IF ((LOBS.LT.2) .OR. (LOBS.GT.NL)) CALL TERMIN(31,0)
        ELSE
        ENDIF
C
C       NOW READ FILTER PARAMETERS
C       -------------------------
C
          READ(1,'(A2)')WWW
          READ(1,*) IPHASE
          IF(IPHASE.EQ.0)THEN
            WRITE(6,'(1X,"FILTER-PHASE")')
            WRITE(6,'(1X,I1,"(ZERO PHASE)")')IPHASE
            MPHASE=.FALSE.
          ELSEIF(IPHASE.EQ.1)THEN
            WRITE(6,'(1X,"FILTER-PHASE")')
            WRITE(6,'(1X,I1,"(MINIMUM PHASE)")')IPHASE
            MPHASE=.TRUE.
          ELSEIF(IPHASE.NE.0.AND.IPHASE.NE.1)THEN
            WRITE(6,'(1X,"NO FILTERS")')
            GO TO 15
          ELSE
          END IF
          WRITE(6,'(//,1X,"FILTER TYPE",2X,"DB SLOPE",2X,
     +    "CUT-OFF FREQUENCY")')
          READ(1,'(A2)')WWW
          DO 10 I=1,15
            READ(1,*,END=15) IFILTYP(I),IDB(I),FF1(I)
            IF(IFILTYP(I).EQ.1)GO TO 11
            IF(IFILTYP(I).EQ.3)GO TO 41
            GO TO 21
  11        WRITE(6,104)IFILTYP(I),IDB(I),FF1(I)
 104        FORMAT(1X,I1,"(LOW PASS)",4X,I3,10X,F4.0)
            GO TO 31
  21        WRITE(6,105)IFILTYP(I),IDB(I),FF1(I)
 105        FORMAT(1X,I1,"(HIGH PASS)",3X,I3,10X,F4.0)
            GO TO 31
  41        WRITE(6,106)IFILTYP(I),IDB(I),FF1(I)
```

```
 106        FORMAT(1X,I1,"(NOTCH FILTER)",3X,I3,10X,F4.0,3X,F4.0)
 31         NFILT=NFILT+1
            F1(NFILT) = FF1(NFILT)*.001
 10       CONTINUE
 15   IF (NFILT .GT. 10) CALL TERMIN(30,NFILT)
      CLOSE(UNIT=1
      RETURN
      END


C-----------------------------------------------------------
C
C              SUBROUTINE EXEMODL
C
C--------------------------------------------------------

      SUBROUTINE EXEMODL
      COMMON / NUM / CL(200),QL(200),RHO(200),T(200),EPS,SIGMA,
     +               CRUST,NL,NTRACE,LOBS,NW,TSEC,TSC,ASC,NMULT,
     +               TLAG,YSIDE,TLNTH
      COMMON / CHAR / ITIME,IDATE,IOBT,TITLE
      COMMON /FILTDT/ F1(10),IDB(10),IFILTYP(10),NFILT,MPHASE,
     + FF1(10)
      CHARACTER*1 IOBT
      CHARACTER*10 ITIME,IDATE
      CHARACTER*80 TITLE
      COMPLEX GL1,GL2
      COMPLEX EIW,WIGH,RCOEFF,C0,C1,AL1,AL2
      COMPLEX EWIGH,RVRB1,RVRB2
      COMPLEX EWRUN0,EWRD,RVRB1EW,RVRB2EW
      COMPLEX RTB1,RTB2,RTB3,TT0NP,TUN0P
      COMPLEX RD0NP,RUN0P,RD0N,RUN0
      COMPLEX TD0N,TUN0,RD,RU,TD,TU
      COMPLEX MU,MD,RUR0,TD0R
      COMPLEX PRESSFA,PR,UNIT,REFW(1024)
      COMMON  /TS/ REFT(1024)
      DATA    PI/3.141592653/
      DATA    BEX/-20.0/
      DATA    C0/(0.0,0.0)/, C1/(1.0,0.0)/
C
C          INITIALIZE BEGINNING OF PLOT
C          --------------------------
C
      NT = 2*NW
      IF (TSEC .GT. 150.0) THEN
        TLAG = 10.0
      ELSEIF (TSEC .GT. 90.0) THEN
        TLAG = 5.0
      ELSE
        TLAG = 1.0
      ENDIF
      UNIT  = C1
C
C         LOOP OVER ALL FREQUENCIES
C         -----------------------
```

```
C
      DO 10 IW=1, NW-1
        W   = IW*PI*2/TSEC
        EIW = CMPLX(EPS,-W)**SIGMA
        AL1 = CMPLX(1.0/CL(1),0.0)*(CMPLX(0.5/QL(1),0.0)/EIW+C1)
        RHO1 = RHO(1)
        CALL PARMGEN(AL1,GL1)
C
C         LOOP OVER ALL LAYERS
C         --------------------
C
        DO 15 IL=1, NL-1
          AL2 = CMPLX(1.0/CL(IL+1),0.) *
     +          (CMPLX(0.5/QL(IL+1),0.0)/EIW + C1)
          RHO2 = RHO(IL+1)
          CALL PARMGEN(AL2,GL2)
          CALL REFL(RHO1,RHO2,GL1,GL2,RU,RD,TU,TD)
          IF (IL .EQ. 1) THEN
            RD0N=RD
            RUN0=RU
            TD0N=TD
            TUN0=TU
            MD = C0
            MU = C0
          ELSE
            WIGH = CMPLX(0.0,W*T(IL))*GL1
            IF (REAL(WIGH) .GT. BEX) THEN
              EWIGH = CEXP(WIGH)
            ELSE
              EWIGH = C0
            ENDIF
            EWRD=EWIGH*RD
            EWRUN0=EWIGH*RUN0
            RTB1=EWRUN0*EWRD
            IF (NMULT .LT. 0) THEN
C
C           ...COMPUTE ALL MULTIPLES RVRB2=1+EWIGH*RD*RVRB1*EWIGH*RUNO.
C
              CALL REVERB(RTB1,RVRB1)
              RTB1=RVRB1*EWRUN0
              RTB2=EWRD*RTB1
              RVRB2=UNIT+RTB2
            ELSE
C
C           ...VALUE FOR NO MULTIPLES.
C
              RVRB1=UNIT
              RVRB2=UNIT
              IF (NMULT .GT. 0) THEN
C
C               ...COMPUTE FINITE NUMBER OF MULTIPLES.
C
C
                DO 20 I=1, NMULT
                  RTB2=RVRB1*RTB1
```

```
            RVRB1=UNIT+RTB2
  20      CONTINUE
           RTB1=EWRD*EWRUN0
           DO 25 I=1, NMULT
            RTB2=RVRB2*RTB1
            RVRB2=UNIT+RTB2
 25       CONTINUE
         ENDIF
        ENDIF
        RTB1=EWIGH*TD0N
        RVRB1EW=RVRB1*RTB1
C
C       ...TD0N=TD*RVRB1*EWIGH*TD0N.
C
        TD0NP=TD*RVRB1EW
        RTB1=EWIGH*TU
        RVRB2EW=RVRB2*RTB1
C
C       ...TUN0=TUN0*RVRB2*EWIGH*TU
C
        TUN0P=TUN0*RVRB2EW
        RTB1=EWRD*RVRB1EW
C
C       ...RD0N=RD0N+TUN0*EWIGH*RD*RVRB1*EWIGH*TD0N.
C
        RTB2=TUN0*RTB1
        RD0NP=RD0N+RTB2
        RTB1=EWRUN0*RVRB2EW
C
C       ...RUN0=RU+TD*EWIGH*RUN0*RVRB2*EWIGH*TU.
C
        RTB2=TD*RTB1
        RUN0P=RU+RTB2
        TD0N=TD0NP
        TUN0=TUN0P
        RD0N=RD0NP
        RUN0=RUN0P
      ENDIF
      TEST = CABS(TD0N)
C
C     ...EXIT LAYER LOOP
C
      IF((TEST.EQ.0.0).OR.(ALOG10(TEST).LT.-8))GO TO 30
      IF (IL .EQ. LOBS-1) THEN
        RUR0=RUN0
        TD0R=TD0N
        TD0N=UNIT
        TUN0=UNIT
        RUN0 = C0
        RD0N = C0
        PRESSFA = RHO(LOBS)*(1.0-(4.0/3.0))
C
C      ...THE OBS. SITS IN THE UPPERMOST PART.
C
        MU = -GL2
```

```
            MD = GL2
          ENDIF
          AL1   = AL2
          RHO1  = RHO2
          GL1   = GL2
C
C        ...END OF THE LAYER LOOP.
C
 15     CONTINUE
 30     CONTINUE
        IF (LOBS .GT. 1) THEN
C
C        ...IF AN OBS MOTION IS ASKED FOR THE INCIDENT PRESSURE WAVE
C        ...IS A DERIVATIVE OF A DELTA FUNCTION.  IF AN OBS PRESSURE
C        ...IS ASKED FOR, THE INCIDENT PRESSURE WAVE IS ASSUMED TO BE
C        ...A DELTA FUNCTION.
C
          RTB1=RUR0*RD0N
      IF(NMULT.LT.0) THEN
         CALL REVERB(RTB1,RTB2)
      ELSE
         RTB2=UNIT
      IF(NMULT.GT.0) THEN
         DO 40 I=1,NMULT
            RTB2=UNIT+RTB1*RTB2
40    CONTINUE
      ENDIF
      ENDIF
C
          RTB1=RTB2*TD0R
          RTB2=RD0N*RTB1
          PR=PRESSFA*(RTB2+RTB1)
          RTB2=MU*RD0N
          RTB2=MD+RTB2
          RTB3=RTB2*RTB1
C
C         ADJUST FOR RECEIVER TYPE
C         ------------------------
C
          IF(IOBT.EQ.'V')THEN
            RCOEFF = RTB3/RHO(1)
          ELSE
C
C         ...PRESSURE.
C
            RCOEFF = PR/RHO(1)
          ENDIF
        ELSE
          RCOEFF = RD0N
        ENDIF
        RCOEFF = RCOEFF * CMPLX(50.*(1+COS(PI*IW/NW)),0.) *
     +               CEXP(CMPLX(0.0,W*TLAG))
        REFW(IW+1) = RCOEFF
C
C        ... NEGATIVE FREQUENCIES ARE COMPLEX CONJ.
```

```
C
           REFW(NT-IW+1) = CONJG(RCOEFF)
C
C       ...END FREQUENCY LOOP
C
 10     CONTINUE
C
C       ...DC AND NYQUIST FREQUENCY SET = TO 0
C
        REFW(1) = C0
        REFW(NW+1) = REFW(1)
        CALL FILTER(REFW,NW,TSEC)
C
C       ...TRANSFORM TO TIME DOMAIN
C
        CALL FFT(NT,REFW,-1.)
        DO 35 I=1, NT
            REFT(I) = REAL(REFW(I))
 35     CONTINUE
        RETURN
        END


C---------------------------------------------------------------
C
C                 SUBROUTINE PARMGEN
C
C---------------------------------------------------------------

        SUBROUTINE PARMGEN(AL,GL)
        COMPLEX AL,GL
        GL = CSQRT(AL*AL)
        IF (AIMAG(GL).LT.0.0) GL=-GL
        RETURN
        END


C-------------------------------------------------------
C
C                    SUBROUTINE REFL
C
C-------------------------------------------------------

        SUBROUTINE REFL(RHO1,RHO2,GL1,GL2,RU,RD,TU,TD)

C         ...COMPUTES REFLECTION COEFFICIENT RU,RD,TU & TD FOR GIVEN LAYER
C         ...INTERFACE PARAMETERS.   THE REFLECTION/TRANSMISSION COEFFICIENTS ARE
C         ...RATIOS OF DISPLACEMENT POTENTIALS.  IN THE KTH LAYER, THE DISPLACEMENT
C         ...VECTOR U(K) IS GIVEN BY:
C            U(K) = GRAD(PHIU(K)+PHID(K))/(I*W) +
C                            CURL CURL(ZHAT(PSIU(K)+PSID(K))) /(W*W*P)
C         ...IN WHICH PHIU AND PHID ARE THE POTENTIALS ASSOCIATED WITH THE UP AND
C         ...DOWN GOING P-WAVES RESPECTIVELY. ZHAT IS A UNIT VECTOR
C         ...POINTING IN THE POSITIVE Z DIRECTION, I.E. DOWNWARDS.
C         ...THEN, FOR EXAMPLE, THE DOWNWARD TRANSMISSION COEFFICIENT FOR P WAVES
```

```
C        ...FROM LAYER 1 TO LAYER 2 IS  TPP12 = PHID(2)/PHID(1).
C
      COMPLEX GL1,GL2
      COMPLEX RU,RD,TU,TD,C0
      COMPLEX Y
C
C       ...NOTE THAT:  GL=1/ALPHA
C
      DATA C0/(0.0,0.0)/
          Y = RHO1*GL2 + RHO2*GL1
          TD = 2.0*RHO1*GL1/Y
          RD = (RHO2*GL1 - RHO1*GL2)/Y
          TU = 2.0*RHO2*GL2/Y
          RU = -RD
      RETURN
      END


C-----------------------------------------------------
C
C               SUBROUTINE REVERB
C
C-----------------------------------------------------

      SUBROUTINE REVERB(A,B)
      COMPLEX A,B,X,Y,DET,C0,C1
      LOGICAL WARNED
      DATA C0/(0.0,0.0)/,C1/(1.0,0.0)/,WARNED/.FALSE./
      IF (ABS(REAL(A))+ABS(AIMAG(A)).LT. 1.0E-12)THEN
        IF (.NOT. WARNED)THEN
          WRITE(6,'(/,"***MULTIPLES IGNORED WHERE MAGNITUDE < E-12")')
          WARNED = .TRUE.
        ENDIF
C
C       ...IGNORE MULTIPLES, TO AVOID UNDERFLOW PROBLEMS
C
        B = C1
      ELSE
        X = 1.0 - A
        Y = 1.0
        DET = X*Y
        B = Y/DET
      ENDIF
      RETURN
      END



C-----------------------------------------------------
C
C           SUBROUTINE FILTER
C
C-----------------------------------------------------
```

```
      SUBROUTINE FILTER(REFW,NW,TSEC)
      LOGICAL MPHASE
      COMMON /FILTDT/ F1(10),IDB(10),IFILTYP(10),NFILT,MPHASE,
     + FF1(10)
      REAL        MAXABSV
      LOGICAL     FRSTTIM
      COMPLEX     FILT(1024), FILT1(1024), REFW(1024)
      DATA        PI/3.141592/
      DATA        FRSTTIM/.TRUE./
      IF (NFILT .EQ. 0) RETURN
C
C       .....GENERATE FILTER.....
C
      IF (FRSTTIM) THEN
C
C     ...INITIATE FILTER IN POSITIVE FREQUENCY DOMAIN. NEGATIVE
C     ...FREQUENCIES BECOME COMPLEX CONJUGATES OF POSITIVE FREQUENCIES
C     ...AND CAN BE COMPUTED WHEN NECESSARY.
C
        DO 10 I=1, NW+1
          FILT(I) = CMPLX(1.0, 0.0)
 10     CONTINUE
C
C       ...SMEARING RANGE
C
      DELFREQ = 0.0
        IF (MPHASE) DELFREQ=2*PI/TSEC
        DO 15 I=1, NFILT
          IF (IFILTYP(I) .EQ. 1)  THEN
C
C         ...LOW PASS BUTTERWORTH FILTER
C
            POLES = FLOAT(IDB(I))/6 + 1
            R = 1/((F1(I)-DELFREQ)*TSEC)
            DO 20 J=1, NW+1
              FILT(J) = FILT(J)/SQRT(1+((J-1)*R)**POLES)
 20         CONTINUE
          ELSEIF (IFILTYP(I) .EQ. 2) THEN
C
C         ...HIGH PASS BUTTERWORTH FILTER
C
            POLES = FLOAT(IDB(I))/6 + 1
            FACTOR = 1/((F1(I)+DELFREQ)*TSEC)
            DO 25 J=1, NW+1
              R = ((J-1)*FACTOR)**POLES
              FILT(J) = FILT(J)*SQRT(R/(1+R))
 25         CONTINUE
          ELSE
C
C         ...NOTCH FILTER
C
            IF(I.EQ.2)GO TO 26
            INDEX1 = IFIX((F1(I)-DELFREQ)*TSEC) + 1
            GO TO 27
   26       INDEX2 = IFIX((F1(I)+DELFREQ)*TSEC) + 2
```

```
            IF (INDEX2-INDEX1 .LT. 16) THEN
              WRITE(6,'(" *** F1(1) TOO CLOSE TO F1(2) IN NOTCH - NO "
     +                   "FILTERING DONE ***")')
              RETURN
            ENDIF
C
C           ...IDB REDUCTION
C
            GO TO 29
   27       R = 1 - 10**(-(FLOAT(IDB(I))/20))
C
C           ...NEILS DESIGN
            IF(I.EQ.1)GO TO 28
C
   29       DO 30 J=INDEX1, INDEX2
              FILT(J) = FILT(J)*
     +          (1 + R*(COS(2*PI*(J-INDEX1)/(INDEX2-INDEX1))-1)/2)
   30       CONTINUE
   28     ENDIF
 15     CONTINUE
        IF (MPHASE) THEN
C
C       ...SET A(W) = LN F(W)
C
          DO 35 I=1, NW+1
            IF (REAL(FILT(I)) .EQ. 0.0) THEN
              FILT(I) = CMPLX(-30.0, 0.0)
            ELSE
              FILT(I) = CMPLX(ALOG(REAL(FILT(I))), 0.0)
            ENDIF
 35       CONTINUE
C
C         ... NEGATIVE NEEDED FOR FFT
C
          DO 40 I=NW+2, NW*2
            FILT(I) = CONJG(FILT(NW*2-I+2))
 40       CONTINUE
C
C         ...A(T) = INVERSE FFT A(W)
C
          CALL FFT(NW*2,FILT,-1.0)
C         ... B(T) = A(T)*G(T)
          DO 45 I=2, NW+1
C         ... G(T)
            R = (1+COS(PI*(I-1)/NW)) /2
C           ... NW+1 VALUE = 0
            FILT(I) = R*FILT(I)
C           ...1 ST VALUE UNCHANGED
            FILT(NW*2+2-I) = R*FILT(NW*2+2-I)
C           ...2 & NW*2 VALUE NEARLY = 0
 45       CONTINUE
C         ...NW+1 VAL = 0 IS USED HERE
          DO 50 I=1, NW
            FILT1(I) = FILT(I)
 50       CONTINUE
```

```
C          ... C(T) HAS INVERTED 2 HALF
           DO 55 I=NW+1, NW*2
             FILT1(I) = -CONJG(FILT(I))
  55       CONTINUE
C          ... B(W) = FT B(T),  C(W) = FT C(T)
           CALL FFT(NW*2,FILT,1.0)
           CALL FFT(NW*2,FILT1,1.0)
           MAXABSV = 0.0
C          ... F'(W) = EXP(B(W)+C(W))
           DO 60 I=1, NW+1
             FILT(I)=CEXP(FILT(I) + FILT1(I))
             MAXABSV = AMAX1(CABS(FILT(I)),MAXABSV)
 60        CONTINUE
           DO 65 I=1, NW+1
C          ...NORMALIZE
             FILT(I) = FILT(I)/MAXABSV
 65        CONTINUE
         ENDIF
         FRSTTIM = .FALSE.
       ENDIF
C
C      .....ACTUAL FILTERING FOLLOWS.....
C
C      ...MULTIPLY BY POSITIVE FREQ FILTER
C
       DO 70 I=1, NW+1
         REFW(I) = REFW(I) * FILT(I)
 70    CONTINUE
C
C      ... NEGATIVE FREQUENCY ARE CONJUGATES
C
       DO 75 I=NW+2, NW*2
         REFW(I) = CONJG(REFW(NW*2-I+2))
 75    CONTINUE
       RETURN
       END


C--------------------------------------------------------
C
C          SUBROUTINE FFT
C
C--------------------------------------------------------

       SUBROUTINE FFT(LX,CX,SIGNI)
       COMPLEX CX(LX),CTEMP,CW
       J = 1
       SC = SQRT(1./FLOAT(LX))
       DO 3 I=1,LX
       IF(I.GT.J) GO TO 1
       CTEMP = CX(J)*SC
       CX(J) = CX(I)*SC
       CX(I) = CTEMP
  1    M = LX/2
  2    IF(J.LE.M) GO TO 3
```

```
      J = J - M
      M = M/2
      IF(M.GE.1) GO TO 2
  3   J = J + M
      L = 1
  4   ISTEP = 2*L
      DO 5 M=1,L
      AA = 314159265358979.D-14*SIGNI*(M-1)/L
      CW = CMPLX(COS(AA),SIN(AA))
      DO 5 I=M,LX,ISTEP
      CTEMP = CW*CX(I+L)
      CX(I+L) = CX(I) - CTEMP
  5   CX(I) = CX(I) + CTEMP
      L = ISTEP
      IF(L.LT.LX) GO TO 4
      RETURN
      END


C---------------------------------------------
C
C         SUBROUTINE WRTPLT
C
C---------------------------------------------

      SUBROUTINE WRTPLT
      COMMON / NUM / CL(200),QL(200),RHO(200),T(200),EPS,SIGMA,
     +                CRUST,NL,NTRACE,LOBS,NW,TSEC,TSC,ASC,NMULT,
     +                TLAG,YSIDE,TLNTH
      COMMON  /TS/ REFT(1024)
      COMMON/AR/IAGC,WINDOW
      COMMON /FILTDT/ F1(10),IDB(10),IFILTYP(10),NFILT,MPHASE,
     + FF1(10)
      COMMON / CHAR /ITIME,IDATE,IOBT,TITLE
      CHARACTER*1 IOBT
      CHARACTER*10 ITIME,IDATE
      CHARACTER*80 TITLE
      OPEN(UNIT=4,FILE='PLTIN',STATUS='NEW')
      IF(IOBT.EQ.'P')THEN
        INUM=1
      ELSE IF(IOBT.EQ.'V')THEN
        INUM=2
      ELSE
        INUM=3
      END IF
      NT=2*NW
      WRITE(4,1001)TITLE
 1001 FORMAT(A)
      WRITE(4,999)NTRACE,YSIDE,NW,TLNTH,NL,NFILT,MPHASE
 999  FORMAT("NTRACE=",I4,",YSIDE=",F5.2,",NW=",I5,",TLNTH=",
     +F5.2,",NL=",I3,",NFILT=",I3,",MPHASE=",L2)
      WRITE(4,989)TSEC,TSC,TLAG,ASC
 989  FORMAT("TSEC=",F10.0,",TSC=",F10.5,",TLAG=",F10.0,",ASC=",F10.5)
      DO 5 I=1,NFILT
```

```
      WRITE(4,995)IFILTYP(I),IDB(I),FF1(I)
  995 FORMAT("FILTER TYPE=",I2,1X,"DB-SLOPE=",I3,1X,
     *"CUT-OFF FREQUENCY=",F5.1)
    5 CONTINUE
      WRITE(4,987)INUM,LOBS,NMULT,CRUST,IAGC,WINDOW
 987  FORMAT(1X,"IOBT=",I1,",LOBS=",I3,",NMULT=",I5,",CRUST=",F10.5
     +,",IAGC=",I3,",WINDOW",F7.2)
      DO 6 I=1,NL
        WRITE(4,'(1X,"CL=",F5.2,",QL=",F6.0,",RHO=",F5.2,",T=",F6.1)')
     +CL(I),QL(I),RHO(I),T(I)
 6    CONTINUE
      DO 10 I=1,NT
        WRITE(4,'("REFT=",1X,F20.14)')REFT(I)
 10   CONTINUE
      RETURN
      END


C-----------------------------------------------------
C
C             SUBROUTINE AGCFREQ
C
C-----------------------------------------------------

      SUBROUTINE AGCFREQ
      DIMENSION SQREFT(1024),BB(2)
      COMPLEX SQREFW(1024), WW(1024)
      COMMON / NUM / CL(200),QL(200),RHO(200),T(200),EPS,SIGMA,
     +               CRUST,NL,NTRACE,LOBS,NW,TSEC,TSC,ASC,NMULT,
     +               TLAG,YSIDE,TLNTH
      COMMON / CHAR / ITIME,IDATE,IOBT,TITLE
      COMMON  /TS/ REFT(1024)
      COMMON/AR/IAGC,WINDOW
      DATA PI/3.141592653/
C
C     ....SQUARE TRACE AMPLITUDES AND MAKE COMPLEX
C     ....INITIALIZE FILTER TO COMPLEX ZERO
C
      SUMSQ=0.0
      NT = NW*2
      DO 330 I = 1,NT
        SQREFT(I) =REFT(I)*REFT(I)
        SUMSQ=SUMSQ+SQREFT(I)
        SQREFW(I) =CMPLX(SQREFT(I),0.0)
        WW(I) = CMPLX(0.0,0.0)
 330  CONTINUE
C
C         CONDITION THE INPUT TIME SERIES
C
      AVSQ=SUMSQ/NT
      BOOST=1.0/AVSQ
      DO 331 I=1,NT
         SQREFW(I) = SQREFW(I)*BOOST
331   CONTINUE
```

```
C
C     ....CONSTRUCT AGC FILTER IN TIME DOMAIN
C     ...."WINDOW" (READ IN BY USER) IS FILTER LENGTH IN MSECS
C
      DELTAT = TSEC/NT
      ITW2 = NINT((WINDOW/DELTAT)/2.0)
      DO 340 I = 1,ITW2+1
        WW(I) =CMPLX(0.5*(1.0+COS(PI*(I-1)/ITW2)),0.0)
        J = NT+1-I
        WW(J) = WW(I)
 340  CONTINUE
C
C             CONDITION THE FILTER
C
      SUMFIL=0.0
      DO 341 I=1,NT
         SUMFIL=SUMFIL + WW(I)
341   CONTINUE
      BOST = NT/SUMFIL
      DO 342 I=1,NT
         WW(I)=WW(I)*BOOST
342   CONTINUE
C
C     ...TRANSFORM TO FREQUENCY DOMAIN AND FILTER
C
      CALL FFT(NT,SQREFW,+1.0)
      CALL FFT(NT,WW,+1.0)
      DO 350 I =1,NT
        SQREFW(I) = SQREFW(I)*WW(I)
 350  CONTINUE
C
C   ...TRUNCATE DC COMPONENT TO 5 SIGNIFICANT FIGURES
C
      TEMP=REAL(SQREFW(1))
      ENCODE(14,345,BB)TEMP
      DECODE(14,345,BB)TEMP1
345   FORMAT(1E14.5)
      SQREFW(1) = CMPLX(TEMP1,0.0)
C
C   ...RETURN TO TIME DOMAIN
C
      CALL FFT(NT,SQREFW,-1.0)
C
C             RESTORE TIME SERIES
C
      SUMSQF = 0.0
      DO 346 I=1,NT
         SUMSQF = SUMSQF + SQREFW(I)**2
346   CONTINUE
      AVSQF = SUMSQF/NT
      BOOST = AVSQ/AVSQF
      DO 347 I=1,NT
         SQREFW(I) = SQREFW(I)*BOOST
347   CONTINUE
C
```

```
C            APPLY AGC
C
      CURRMIN = REAL(SQREFW(1))
      DO 360 I= 2,NT
          CURRMIN = AMIN1(CURRMIN,REAL(SQREFW(I)))
 360  CONTINUE
      DO 380 I = 1,NT
          IF (CURRMIN.LT.0.0) THEN
             REFT(I) = REFT(I)/SQRT(REAL(SQREFW(I)) + ABS(CURRMIN*2.0))
          ELSE
             REFT(I) = REFT(I)/SQRT(REAL(SQREFW(I)))
          ENDIF
 380  CONTINUE
      RETURN
      END
C-----------------------------------------------------------------
C
C               SUBROUTINE TERMIN
C
C-------------------------------------------------------------------
      SUBROUTINE TERMIN(NERR, LNR)


      COMMON / NUM / CL(200),QL(200),RHO(200),T(200),EPS,SIGMA,
     +               CRUST,NL,NTRACE,LOBS,NW,TSEC,TSC,ASC,NMULT,
     +               TLAG,YSIDE,TLNTH
      COMMON / CHAR / ITIME,IDATE,IOBT,TITLE

      IF (NERR .LT. 10) THEN
        WRITE(6,'(/" *** ERROR IN MODEL FILE LAYER: ",I4)') LNR
      ELSEIF (NERR .LT. 20) THEN
        CALL PRTMLDA
        WRITE(6,'(/" *** ERROR IN PROCESSED LAYER: ",I4)') LNR

      ENDIF

      IF (NERR .EQ. 1) THEN
        WRITE(6,'(" TOO MANY LAYERS IN MODEL: ",I6,", MAX=200")') NL
      ELSEIF (NERR .EQ. 3) THEN
        WRITE(6,'("CL CANNOT BE 0 AND H CANNOT BE -VE FOR FIRST
     +LAYER")')
      ELSEIF (NERR .EQ. 4) THEN
        WRITE(6,'("LAST LAYER CANNOT BE AN INTERPOLATED LAYER")')
      ELSEIF (NERR .EQ. 5) THEN
      WRITE(6,'("ABSOLUTE DEPTH GIVEN IS LESS THAN THE CURRENT DEPTH")')

      ELSEIF (NERR .EQ. 11)THEN
        WRITE(6,'("CL NOT IN RANGE 0 ¡ CL ¡ 10")')
      ELSEIF (NERR .EQ. 12)THEN
        WRITE(6,'("QL ¡ 100")')
      ELSEIF (NERR .EQ. 15) THEN
        WRITE(6,'("RHO NOT IN RANGE 0 ¡ RHO ¡ 10")')
      ELSEIF (NERR .EQ. 21) THEN
        WRITE(6,'(/" *** TOO FEW (OR TOO MANY) VALUES ON ONE LINE "
     +    "OF MODEL FILE -"/" CHECK FOR / AT END OF INTERPOLATED LAYER ",
```

Test 3. Multiples

Two layers, 75 and 100 m thick, are sandwiched between half spaces (fig. 3B). Q = 200 and density = 1.1 for all units. The 100-meter-thick layer has a velocity of 3.0 km/sec; the other units have a velocity of 1.5 km/sec. The seismogram consisted of the time-domain reflection coefficient plotted without filtering.

A series of tests were run with options beginning with "primaries only" to options for a selected number of "multiples" and finally a "complete-response" seismogram (fig. 5). R1 and R2 identify the reflections from the top of the two reflectors. M1 through M4 identify the "peg-leg" multiples. Figure 5A demonstrates the complete-response option; the clipping occurs because of a large amplification option (ASC = 100). Figure 5B shows the seismogram for the same model but for a user option of only one multiple. Note that no reflections are observed beyond the first multiple arrival at 234 ms. Arrival times, amplitudes, and polarity in figure 5 are consistent with those predicted by hand calculations based on reflection-transmission coefficients. Table 3 summarizes the results.

Table 3. Arrival times of reflections and multiples for a two-layer model (fig. 5)

Reflector index	Primary only	One multiple	Two multiples	Full response
R1	100 ms	100 ms	100 ms	100 ms
R2	167 ms	167 ms	167 ms	167 ms
M1	—	234 ms	234 ms	234 ms
M2	—	—	300 ms	300 ms
M3	—	—	—	367 ms
M4	—	—	—	434 ms

Table 4. Input data used to test a full response (fig. 5A)

```
TEST 3 - D2L.  FULL RESPONSE OPTION.
EPS      SIGMA      YSIDE     XSIDE
0.001     0.1        40.       6.
CL         QL        RHO     THICKNESS
1.50       200.      1.1            0.
1.50       200.      1.1           75.
3.00       200.      1.1          100.
1.5        200.      1.1            0.
9999      9999      9999         9999
#TRC  #FREQ  T(MSEC)    IN/MSEC AMP-SCALE MLTPL   AGC   WINDOW
 5     512    1000.       .006    100.0     -1      0      0
DETECTOR TYPE     DETECTOR LOCATION
 REFLECTION           LAYER 1
FILTER PHASE
2  (NO FILTERS)
FILTER-TYPE   DB SLOPE   CUT-OFF FREQUENCY
```

TEST 2 - D1L. HIGH Q (=1500)
REFLECTION COEFFICIENT
AT OBS IN LAYER 1 AT .0 M DEPTH
FREQ(NW)= 512 T(MSECS)=2000
MIN = .50 HZ MAX =256.00 HZ
PRIMARIES ONLY
NO AGC. ASC = 1.
NO FILTERS

TEST 2 - D1L. LOW Q (=5)
REFLECTION COEFFICIENT
AT OBS IN LAYER 1 AT .0 M DEPTH
FREQ(NW)= 512 T(MSECS)=2000
MIN = .50 HZ MAX =256.00 HZ
PRIMARIES ONLY
NO AGC. ASC = 1.
NO FILTERS

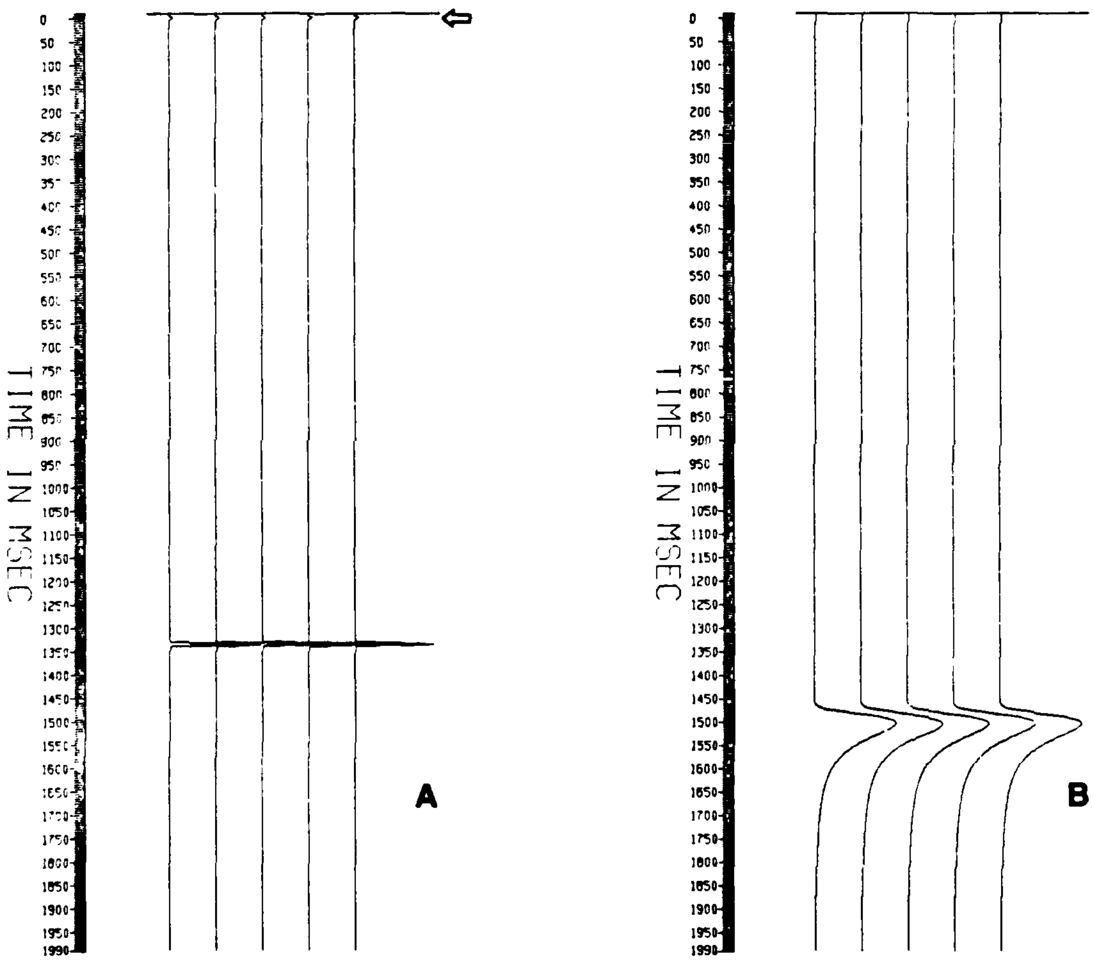


Figure 4. (Test 2) Reflection from a single layer (fig. 3A) for A, Q = 1500 and B, Q = 5. Arrow identifies reflection from change in Q at upper half space.

Test 2. Attenuation Factor Q

A layer 1000 m thick was overlain by a half space with an identical velocity (1.5 km/sec) and underlain by a half space with a velocity of 3.0 km/sec (fig. 3A). All units had a constant density of 1.1 gm/cc. No filter was applied. The seismograms consisted of a plot of reflection coefficients for primary arrivals only. A wide frequency range of 1-512 hz was invoked to study the waveforms better.

The Q of rocks for seismic energy is of the order 50, although poorly consolidated materials may be lower. In this test Q was varied from 5 to 2000. For Q values above 50, the response was essentially a spike with a width of less than 10 msec. Figure 4A illustrates a Q of 1500. For Q values of 20 or less, the pulse width of the response increased to 20 msec, with a width of about 150 msec for a Q of 5 (fig. 4B). As expected, low Q emphasized lower frequencies.

A change of Q from 15 in the top half space to 1500 in a layer 1,000 m thick (but with no change in the user-specified velocity between these two regions) yielded a small pulse at zero time. (See arrow in fig. 4A.) This is consistent with our algorithm, which brings in attenuation changes through a complex velocity.

The following records (table 2) were used as input to Program VISP to generate figure 4A. The same records were used to generate figure 4B, except that QL values were all set equal to five.

Table 2. Input data used to test high Q (fig. 4A)

```
TEST 2 - D1L.  HIGH Q (=1500)
EPS      SIGMA    YSIDE    XSIDE
0.001    0.1      40.      6.0
CL       QL       RHO      THICKNESS
1.5      15.      1.1          0.
1.5      1500.    1.1       1000.
3.0      1500.    1.1          0.
9999     9999     9999      9999
#TRC  #FREQ  T(MSEC)   IN/MSEC  AMP-SCALE  MLTPL  AGC  WINDOW
 5     512   2000.      .003      1.0        0     0     0
DETECTOR TYPE    DETECTOR LOCATION
 REFLECTION          LAYER 1
FILTER PHASE
2  (NO FILTERS)
FILTER-TYPE   DB SLOPE   CUT-OFF FREQUENCY
```

Appendix 4. Input Records and Output Plots for 12 Tests

Thirteen tests were run to demonstrate the reliability of the options available to the user of Program VISP. Test 1 consisted of a model simulating a standard field record (fig. 2). In the remaining 12 tests, three basic models were used: (1) one-layer, (2) two-layer, and (3) six-layer (figs. 3A-3C). Each of the following test cases is related to one of these three models, although the velocity, Q, and (or) density may be varied. Program VISP has detailed comment cards at the beginning that explain the various user options tested in this appendix.

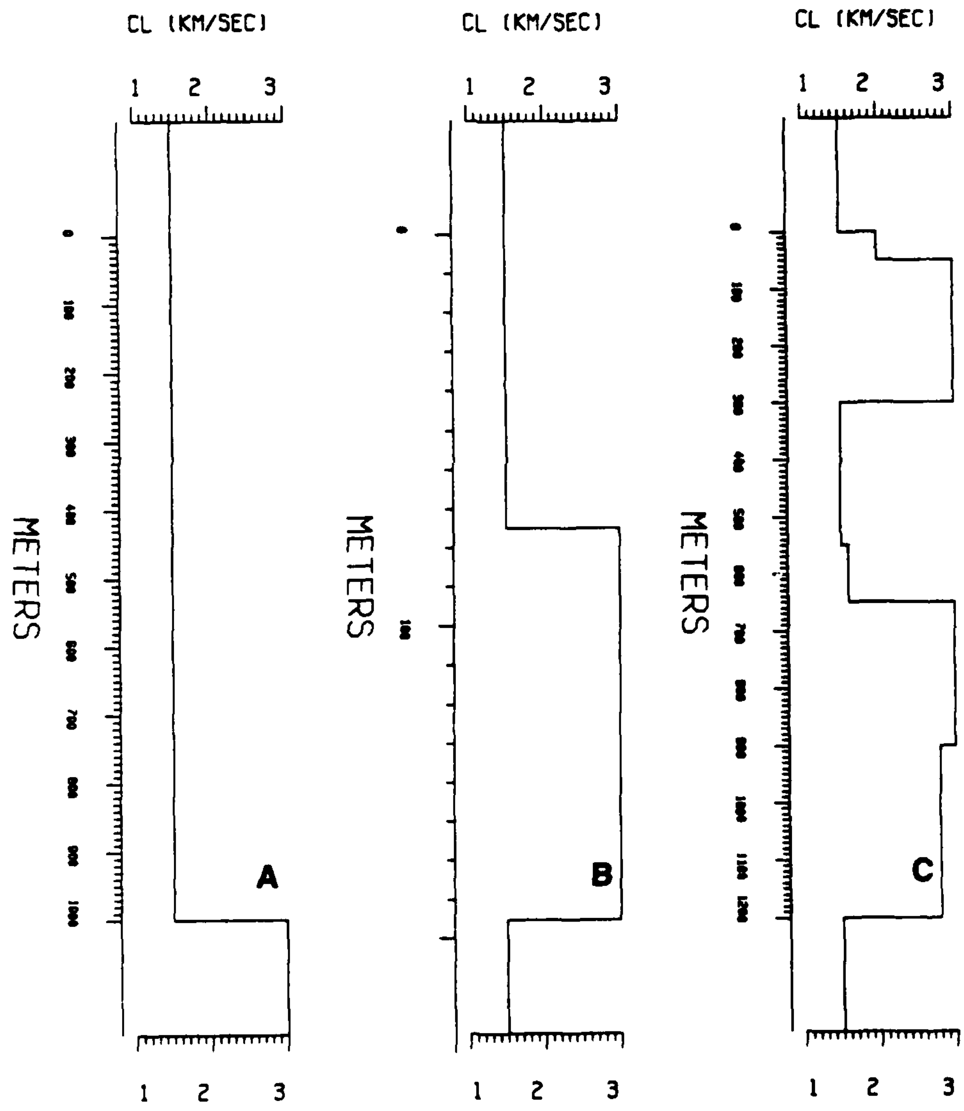


Figure 3. Velocity models for Tests 2-13 (appendix 4). A, one-layer; B, two-layer; C, six-layer.

```
          IF (MOD(IXRANGE,100) .EQ. 0)
     +      CALL NUMBER(X-0.15,Y+YOFFSET,0.055,FLOAT(IXRANGE),0.0,-1)
          CALL PLOT(X,Y,INKOFF)
          CALL PLOT(X+0.75,Y,INKON)
          IF (I.EQ.0) CALL SYMBOL(X0+TLNTH/2-0.4,Y-0.7,0.16,
     +    "METERS",0.0,6)

          Y = Y + 7.0 - 0.4
          X = X0
C
C         PLOT MODEL PARAMETERS
C         ---------------------
C
        DO 170 I=1,3
          IF (PMAX(I) .NE. PMIN(I))GO TO 55
            YSC = 0.0
            Y1  = YSTRIP/2
          GO TO 65
 55         YSC = YSTRIP/(PMAX(I)-PMIN(I))
            Y1  = 0.0
 65       YI = Y0 + 0.2 + (I-1)*(YSTRIP+0.2) + 0.1
          IF ((I.EQ.1) .OR. (I.EQ.5))
     +      CALL PLOT(X0,YI+(PARM(1,I)-PMIN(I))*YSC+Y1,INKOFF)
          X = X0+0.75
          LIMIT=NL-1
          DO 180 J=1,LIMIT
            IF ((J.NE.1) .OR. ((I.LT.2) .OR. (I.GT.4)))GO TO 75
              CALL PLOT(X,YI+(PARM(2,I)-PMIN(I))*YSC+Y1,INKOFF)
            GO TO 85
 75           CALL PLOT(X,YI+(PARM(J,I)-PMIN(I))*YSC+Y1,INKON)
              CALL PLOT(X,YI+(PARM(J+1,I)-PMIN(I))*YSC+Y1,INKON)
 85         IF (J.LT.NL-1) X=X + T(J+1)*(TLNTH-1.5)/IXRANGE
 180      CONTINUE
          X = X0+TLNTH
          CALL PLOT(X,YI+(PARM(NL,I)-PMIN(I))*YSC+Y1,INKON)
          IF(I.EQ.1)CALL SYMBOL(X0-0.6,Y0+(I-1)*(YSTRIP+.2)+.3,.1,
     +"CL (KM/SEC) ",90.,12)
          IF(I.EQ.2)CALL SYMBOL(X0-0.6,Y0+(I-1)*(YSTRIP+.2)+.3,.1,
     +"QL*         ",90.,12)
          IF(I.EQ.3)CALL SYMBOL(X0-0.6,Y0+(I-1)*(YSTRIP+.2)+.3,.1,
     +"RHO (GM/CC) ",90.,12)
          CALL NUMBER(X0-0.6,Y0+1.96,0.10,PARMPY,90.0,3)
 170    CONTINUE
        X0 = X0+TLNTH+2.5
        RETURN
        END
```

```
 10          CALL PLOT(X,Y,INKOFF)
             IF(J.EQ.2)CALL PLOT(X+.1*(I*2-1),Y,INKON)
             CALL PLOT(X+0.1*(I*2-1),Y,INKON)
             CALL NUMBER(X+XOFFSET,Y,0.1,FLOAT(PMIN(J)),90.0,-1)
             CALL PLOT(X,Y,INKOFF)
             ISTART=PMIN(J)+1
             ISTOP=PMAX(J)
             DO 130 K=ISTART,ISTOP
               DO 140 L=1,10
                 Y = Y + YSTRIP/((PMAX(J)-PMIN(J))*10)
                 CALL PLOT(X,Y,INKON)
                 IF(J.NE.2) CALL PLOT(X+0.05*(I*2-1),Y,INKON)
                 CALL PLOT(X,Y,INKOFF)
  140          CONTINUE
               CALL PLOT(X+0.1*(I*2-1),Y,INKON)
                  IF(J.NE.2)
    +               CALL NUMBER(X+XOFFSET,Y-0.1,0.1,FLOAT(K),90.0,-1)
                  IF(J.EQ.2.AND.K.EQ.PMAX(2))
    +               CALL NUMBER(X+XOFFSET,Y-0.1,0.1,FLOAT(K),90.0,-1)
               CALL PLOT(X,Y,INKOFF)
  130        CONTINUE
 20        Y = Y+0.1
  120    CONTINUE
         X = X+TLNTH
         Y = Y0 + 0.2
 110   CONTINUE
C
C        PLOT X AXIS
C        -----------
C
       X = X0
       Y = Y0 + 0.2
       I = 0
         YOFFSET = -0.1+0.25*(I*2-1)
         CALL PLOT(X,Y,INKOFF)
         X = X+0.75
         CALL PLOT(X,Y,INKON)
         DO 160 N=1,IXRANGE,10
           J=N-1
           IF(MOD(J,100).NE.0)GO TO 30
             CALL PLOT(X,Y+0.1*(I*2-1),INKON)
             CALL NUMBER(X-0.05,Y+YOFFSET,0.055,FLOAT(J),0.0,-1)
           GO TO 40
 30         CALL PLOT(X,Y+0.05*(I*2-1),INKON)
 40         CALL PLOT(X,Y,INKOFF)
           X = X + (TLNTH-1.5)/IXRANGE*10
           CALL PLOT (X,Y,INKON)
 160     CONTINUE
         CALL PLOT(X,Y+0.1*(I*2-1),INKON)
```

```
      INKON=2
      INKOFF=3
      Y0=0.5
      X0 = X0+2.0
      PARMAX = ABSPARM(1,2)
      DO 50 I= 2,NL
        PARMAX = AMAX1(PARMAX,ABSPARM(I,2))
   50 CONTINUE
      IF(PARMAX.LT.100.0) PARMPY = 1.0
      IF(PARMAX.GE.100.0.AND.PARMAX.LT.1000.0) PARMPY = 0.1
      IF(PARMAX.GE.1000.0.AND.PARMAX.LT.10000.0) PARMPY = 0.01
      IF(PARMAX.GE.10000.0) PARMPY = 0.001
      DO 60 I=1,NL
        DO 70 J=1,3,2
          PARM(I,J) = ABSPARM(I,J)
   70   CONTINUE
        PARM(I,2) = ABSPARM(I,2)*PARMPY
   60 CONTINUE
      DO 80 I=1,3
        PMAX(I) = 0
        PMIN(I) = 1000
   80 CONTINUE
      PMIN(1) = PARM(1,1)
      DO 90 I=2,NL
        DO 100 J=1,3
          PMAX(J) = MAX1(FLOAT(PMAX(J)),PARM(I,J)+0.999)
          PMIN(J) = MIN1(FLOAT(PMIN(J)),PARM(I,J))
  100   CONTINUE
 90   CONTINUE
      YSTRIP  = 1.0
      IXRANGE = CRUST+0.99
C
C       PLOT Y AXIS
C       -----------
C
      X = X0
      Y = Y0 + 0.2
      DO 110 M=1,2
        I=M-1
        XOFFSET = 0.12+0.3*(I*2-1)
        CALL PLOT(X,Y,INKOFF)
        DO 120 J=1,3
          Y = Y+0.1
          IF(PMIN(J).NE.PMAX(J))GO TO 10
            Y = Y + YSTRIP/2
            CALL PLOT(X,Y,INKOFF)
            CALL PLOT(X+0.1*(I*2-1),Y,INKON)
            CALL NUMBER(X+XOFFSET,Y,0.1,FLOAT(PMIN(J)),90.0,-1)
            Y = Y + YSTRIP/2
          GO TO 20
```

```
C        PLOT TIME SERIES
C        ----------------
C
      CURRMAX = ABS(REFT(1))
        NT=NW*2
          DO 290 I=2,NT
            CURRMAX = AMAX1(CURRMAX,ABS(REFT(I)))
  290     CONTINUE
      X = X0
      Y = Y0 - 2.0
C
C       EFFECTIVELY A NULL TRACE (NOISE)
C
      IF (CURRMAX.LT.0.1) CURRMAX=-10.0
        DO 300 ITHETA=1,NTRACE
          AY = (ITHETA-0.5)*XSC
          CALL PLOT(X,Y+AY,INKOFF)
C
C       ALLOWS FOR TRUNCATED TIME SERIES  (TSC ≠ 0)
C
          NPTS = NW*2*AMIN1(1.0,YSIDE/(TSEC*TSC))
          YSC  = NPTS*TSEC*TSC/((NPTS-1)*NW*2)
            DO 310 I=1,NPTS
              IF(.NOT. ASC.EQ.1.0) GO TO 55
                XX(I) = X+(I-1)*YSC
                YY(I) = Y+AY+REFT(I)/(CURRMAX*2)
                CALL PLOT(XX(I),YY(I),INKON)
             GO TO 310
  55            X1 = AMIN1(ABS(REFT(I))*ASC/(CURRMAX*2), 0.4)
                IF (REFT(I) .LT. 0.0) X1=-X1
                XX(I) = X+(I-1)*YSC
                YY(I) = Y+AY+X1
                CALL PLOT(XX(I),YY(I),INKON)
  310   CONTINUE
  300   CONTINUE
        RETURN
        END
C----------------------------------------------------------
C
C            SUBROUTINE PARMPLT
C
C----------------------------------------------------------
      SUBROUTINE PARMPLT
      COMMON/ PLTPAR /NTRACE,YSIDE,TSEC,TSC,TLAG,ASC,REFT(1024),NW
     +,IOBT,LOBS,NMULT,NL,CL(200),QL(200),RHO(200),T(200),TLNTH,X0
     +,CRUST,IAGC,WINDOW,NFILT
      COMMON/ FILTDT /IDB(10),IFILTYP(10),MPHASE,FF1(10)
      REAL         ABSPARM(200,3), PARM(200,3), TLNTH
      EQUIVALENCE (ABSPARM(1,1),CL(1))
      INTEGER   PMAX(3),PMIN(3)
```

```
C
      X = X0
      Y = Y0 - 2.4
      I=0
        XOFFSET = 0.07+0.3*(I*2-1)
        CALL PLOT(X,Y,INKOFF)
        DO 270 M=1,IYRANGE
          J=M-1
          IF(.NOT.MOD(J-1,MODF) .EQ. INT(TLAG)-1) GO TO 50
            CALL PLOT(X,Y+0.1*(I*2-1),INKON)
C
C CHANGE SIZE OF CHARACTER
C
      IF(TSC.LT.(.01))GO TO 49
      CALL NUMBER(X-0.07,Y+XOFFSET,0.055,FLOAT(J)-TLAG,0.,-1)
      GO TO 50
   49 CALL NUMBER(X+.03,(Y+XOFFSET)-.05,0.055,FLOAT(J)-TLAG,90.,-1)
 50       IF(.NOT.MOD(J,(MODF-1)/10+1).EQ.0) GO TO 51
            CALL PLOT(X,Y+0.07*(I*2-1),INKON)
 51       CALL PLOT(X,Y,INKOFF)
          IF(.NOT.IYRANGE.LE.20)GO TO 52
            DO 280 K=1,10
              X = X + TSC*0.1
              CALL PLOT(X,Y,INKON)
              CALL PLOT(X,Y+0.05*(I*2-1),INKON)
              CALL PLOT(X,Y,INKOFF)
 280        CONTINUE
          GO TO 270
 52         X = X+TSC
            CALL PLOT(X,Y,INKON)
 270    CONTINUE
        CALL PLOT(X,Y+0.1*(I*2-1),INKON)
      IF(TSC.LT.(.01))GO TO 53
        CALL NUMBER(X-0.07,Y+XOFFSET,.055,FLOAT(IYRANGE)-TLAG,0.0,-1)
      GO TO 54
   53 CALL NUMBER(X+.022,(Y+XOFFSET)-.05,.055,FLOAT(IYRANGE)-TLAG,
     *90.,-1)
   54 Y = Y0 - 1.4

      X = X0+IYRANGE*TSC/2-1.5
      CALL SYMBOL(X+.75,Y-1.5,.15,"TIME IN MSEC ",0.0,13)
C
C       PLOT X AXIS
C       -----------
C
      Y = Y0 - 2.0
      CALL PLOT(X0,Y,INKOFF)
      CALL PLOT(X0,Y+0.4+YS,INKON)
C
C
```

```
  33 CALL SYMBOL(X0+1.8,6.5,.1,"---FILTERS = ZERO PHASE---",90.,26)
  34 DO 101 I=1,NFILT
     IF(IFILTYP(I).NE.1)GO TO 35
     CALL NUMBER(X0+1.9+I*.15,7.5,.1,FF1(I),90.,0)
     CALL NUMBER(X0+1.9+I*.15,8.3,.1,FLOAT(IDB(I)),90.,-1)
     CALL SYMBOL(X0+1.9+I*.15,6.5,.1,
    *"HIGH CUT AT        HZ;    DB SLOPE",90.,33)
  35 IF(IFILTYP(I).NE.2)GO TO 36
     CALL NUMBER(X0+1.9+I*.15,7.5,.1,FF1(I),90.,0)
     CALL NUMBER(X0+1.9+I*.15,8.3,.1,FLOAT(IDB(I)),90.,-1)
     CALL SYMBOL(X0+1.9+I*.15,6.5,.1,
    *"LOW  CUT AT        HZ;    DB SLOPE",90.,33)
  36 IF((IFILTYP(I).NE.1).AND.(IFILTYP(I).NE.2))GO TO 37
     GO TO 101
  37 IF(I.EQ.2)GO TO 101
     CALL SYMBOL(X0+1.9+I*.15,6.5,.1,
    *"NOTCH   DB FREQUENCY RANGE      -      HZ",90.,95)
     CALL NUMBER(X0+1.9+I*.15,8.9,.1,FF1(1),90.,-1)
     CALL NUMBER(X0+1.9+I*.15,9.4,.1,FF1(2),90.,-1)
 101 CONTINUE
 31  IF(IAGC.NE.1)GO TO 12
       CALL SYMBOL(X0+1.4,6.5,.1,"AGC WINDOW =         MSECS",90.,24)
       CALL NUMBER(X0+1.4,7.8,.1,WINDOW,90.,0)
     GO TO 30
 12    CALL SYMBOL(X0+1.4,6.5,.1,"NO AGC. ASC =     ",90.,14)
       CALL NUMBER(X0+1.4,7.9,.1,ASC,90.,0)
 30  X0 = X0+3.0
     XSC     = 0.30
C
C     FOR TRUNCATED TIME SERIES
C
     IYRANGE = AMIN1(TSEC,YSIDE/TSC) - 0.01 + 1
C
C     (TSC ≠ 0)
C
     IF (IYRANGE .LE. 10) MODF=1
     IF (IYRANGE.LE.20)MODF=2
     IF(IYRANGE.LE.50)MODF=5
     IF(IYRANGE.LE.100)MODF=10
     IF(IYRANGE.GT.100) MODF=50
     YS  = NTRACE*XSC
C
C      PLOT Y(TIME) AXES
C      --------------------
```

```
      COMMON/ FILTDT /IDB(10),IFILTYP(10),MPHASE,FF1(10)
      DIMENSION XX(1024),YY(1024)
      LOGICAL MPHASE
      INKON=2
      INKOFF=3
      R=0.0
      Y0=9.0
      X0=0.75
      IF(IOBT.EQ.1)CALL SYMBOL(X0,6.5,.1,
     +"PRESSURE                 ",90.,24)
      IF(IOBT.EQ.2)CALL SYMBOL(X0,6.5,.1,
     + "VERTICAL DISPLACEMENT    ",90.,24)
      IF(IOBT.NE.1.AND.IOBT.NE.2)CALL SYMBOL(X0,6.5,
     +.1,"REFLECTION COEFFICIENT  ",90.,22)
      IF(LOBS.NE.0)GO TO 10
      GO TO 15
10    CALL SYMBOL(X0+.26,6.5,.1,
     +"AT OBS IN LAYER        AT         M DEPTH ",90.,39)
      CALL NUMBER(X0+.26,8.1,.1,FLOAT(LOBS),90.,-1)
      CALL SYMBOL(X0+.52,6.5,.1,"FREQ(NW)=",90.,9)
      CALL NUMBER(X0+.52,7.4,.1,NW,90.,-1)
      CALL SYMBOL(X0+.52,8.5,.1,"T(MSECS)=",90.,9)
      CALL NUMBER(X0+.52,9.3,.1,TSEC,90.,-1)
      K=LOBS-1
      IF(K.LT.2) GO TO 11
      DO 20 I=2,K
        R=R+T(I)
20    CONTINUE
11    CALL NUMBER(X0+.26,8.7,.1,R,90.,1)
   15 CALL SYMBOL(X0+.8,6.4,.1," MIN =        HZ  ",90.,17)
      CALL NUMBER(X0+.8,7.2,.1,(1/TSEC)*1000.,90.,2)
      CALL SYMBOL(X0+.8,8.1,.1,"MAX =       HZ",90.,14)
      CALL NUMBER(X0+.8,8.55,.1,(NW/TSEC)*1000.,90.,2)
      IF(NMULT.LT.0)CALL SYMBOL(X0+1.1,6.5,.1,
     +"FULL RESPONSE (ALL MULTIPLES) ",90.,29)
      IF(NMULT.EQ.0)CALL SYMBOL(X0+1.1,6.5,.1,"PRIMARIES ONLY ",
     +90.,14)
      IF(NMULT.GT.0)GO TO 25
      GO TO 28
25    CALL SYMBOL(X0+1.1,6.5,.1,"NUMBER OF MULTIPLES = ",90.,21)
      CALL NUMBER(X0+1.1,8.7,0.1,FLOAT(NMULT),90.,-1)
28    IF(NFILT.GT.0)GO TO 27
      CALL SYMBOL(2.5,6.5,.1,"NO FILTERS",90.,10)
      GO TO 31
   27 IF(.NOT.MPHASE)GO TO 33
      CALL SYMBOL(X0+1.8,6.5,.1,"---FILTERS = MINIMUM PHASE---",90.,29)
      GO TO 34
```

```
 989    FORMAT(5X,F10.0,5X,F10.5,6X,F10.0,5X,F10.5)
        IF(NFILT.EQ.0)GO TO 985
        DO 15 I=1,NFILT
         READ(1,986)IFILTYP(I),IDB(I),FF1(I)
  986    FORMAT(12X,I2,10X,I3,19X,F5.1)
   15   CONTINUE
 985    READ(1,987)IOBT,LOBS,NMULT,CRUST,IAGC,WINDOW
 987    FORMAT(6X,I1,6X,I3,7X,I5,7X,F10.5,6X,I3,7X,F7.2)
        NT=2*NW
        DO 5 I=1,NL
          READ(1,102)CL(I),QL(I),RHO(I),T(I)
 102      FORMAT(4X,F5.2,4X,F6.0,5X,F5.2,3X,F6.1)
 5      CONTINUE
        DO 10 I=1,NT
          READ(1,101)REFT(I)
 101      FORMAT(5X,F20.14)
 10     CONTINUE
        RETURN
        END
C-----------------------------------------------------------------
C
C                          SUBROUTINE INITPLT
C
C-----------------------------------------------------------------
        SUBROUTINE INITPLT
        COMMON/ CHAR /TITLE(8)
        COMMON/ PLTPAR /NTRACE,YSIDE,TSEC,TSC,TLAG,ASC,REFT(1024),NW
       +,IOBT,LOBS,NMULT,NL,CL(200),QL(200),RHO(200),T(200),TLNTH,X0
       +,CRUST,IAGC,WINDOW,NFILT
        COMMON/ FILTDT /IDB(10),IFILTYP(10),MPHASE,FF1(10)
        CALL SYMBOL(.5,.5,0.15,"VISP; ",90.,6)
        CALL SYMBOL(.5,.8,.15,"     PLOTS OF PARAMETERS        ",
       +90.0,34)
        CALL SYMBOL(1.5,.5,.1,TITLE,90.,80)
        CALL SYMBOL(.5,6.5,.12,TITLE,90.,80)
        X0=1.5
        RETURN
        END
C-----------------------------------------------------------------
C
C                          SUBROUTINE RSPMDL
C
C-----------------------------------------------------------------
        SUBROUTINE RSPMDL
        COMMON/ PLTPAR /NTRACE,YSIDE,TSEC,TSC,TLAG,ASC,REFT(1024),NW
       +,IOBT,LOBS,NMULT,NL,CL(200),QL(200),RHO(200),T(200),TLNTH,X0
       +,CRUST,IAGC,WINDOW,NFILT
```

```
C                 ================================
C
C                           PROGRAM PLTVISP
C
C                 (PLOTTING ROUTINE FOR PROGRAM VISP)
C
C          MODIFIED FOR CDC 855 USE AT INDIANA UNIVERSITY
C          FROM PLOTTING CODE DEVELOPED BY FRAZER,BATES
C          AND RUDMAN AT THE HAWAII INSTITUTE OF GEOPHYSICS
C
C          INPUT FOR THIS PROGRAM IS GENERATED AS A FILE
C          ON TAPE 4 (AND DESIGNATED "PLTIN") BY PROGRAM
C          VISP (SEE SUBROUTINE WRTPLT).
C
C-----------------------------------------------------------------------
C
C                             MAIN PROGRAM
C
C-----------------------------------------------------------------------
      PROGRAM TSTPLT(TAPE1,OUTPUT,TAPE4=OUTPUT,PLOT,TAPE7=PLOT)
      COMMON/ CHAR /TITLE(8)
      COMMON/ PLTPAR /NTRACE,YSIDE,TSEC,TSC,TLAG,ASC,REFT(1024),NW
     +,IOBT,LOBS,NMULT,NL,CL(200),QL(200),RHO(200),T(200),TLNTH,X0
     +,CRUST,IAGC,WINDOW,NFILT
      COMMON/ FILTDT /IDB(10),IFILTYP(10),MPHASE,FF1(10)
      CALL IDENT(7)
      CALL READINP
      CALL INITPLT
      CALL PARMPLT
      CALL RSPMDL
      CALL PLOT(0.,0.,999)
      STOP
      END

C-----------------------------------------------------------------
C
C                          SUBROUTINE READINP
C
C-----------------------------------------------------------------
      SUBROUTINE READINP
      COMMON/ CHAR /TITLE(8)
      COMMON/ PLTPAR /NTRACE,YSIDE,TSEC,TSC,TLAG,ASC,REFT(1024),NW
     +,IOBT,LOBS,NMULT,NL,CL(200),QL(200),RHO(200),T(200),TLNTH,X0
     +,CRUST,IAGC,WINDOW,NFILT
      COMMON/ FILTDT /IDB(10),IFILTYP(10),MPHASE,FF1(10)
      READ(1,1001)TITLE
 1001 FORMAT(8A10)
      READ(1,999)NTRACE,YSIDE,NW,TLNTH,NL,NFILT,MPHASE
 999  FORMAT(7X,I4,7X,F5.2,4X,I5,7X,F5.2,4X,I3,7X,I3,8X,L2)
      READ(1,989)TSEC,TSC,TLAG,ASC
```

APPENDIX 3

```
+     "DECLARATION LINES")')

 ELSEIF (NERR .EQ. 23) THEN
   WRITE(6,'(/" *** ERROR - NFREQ ",I4," MUST BE A POWER OF 2"
+                 ", 16-512")') LNR
 ELSEIF (NERR .EQ. 25) THEN
   WRITE(6,'(/" *** ERROR - TIME ",I4," TOO SHORT (MIN 1 MSEC)"
+                 )') LNR
 ELSEIF (NERR .EQ. 27) THEN
   WRITE(6,'(/" *** ERROR - NTRACE ł 10")')
 ELSEIF (NERR .EQ. 29) THEN
   WRITE(6,'(/" *** ERROR - ASC ł 1 OR ł 100")')

 ELSEIF (NERR .EQ. 30) THEN
   WRITE(6,'(/" *** ERROR - ł 10 FILTERS REQUESTED: ",I4)')
+         LNR
 ELSEIF (NERR .EQ. 31) THEN
   WRITE(6,'(/" *** LAYER FOR OBS PLACEMENT ł 2 OR ł NUMBER",
+                 " OF LAYERS")')
 ELSE
   WRITE(6,'(" *** UNKNOWN ERROR, NUMBER: ",I4)') NERR
 ENDIF
 STOP
 END
```

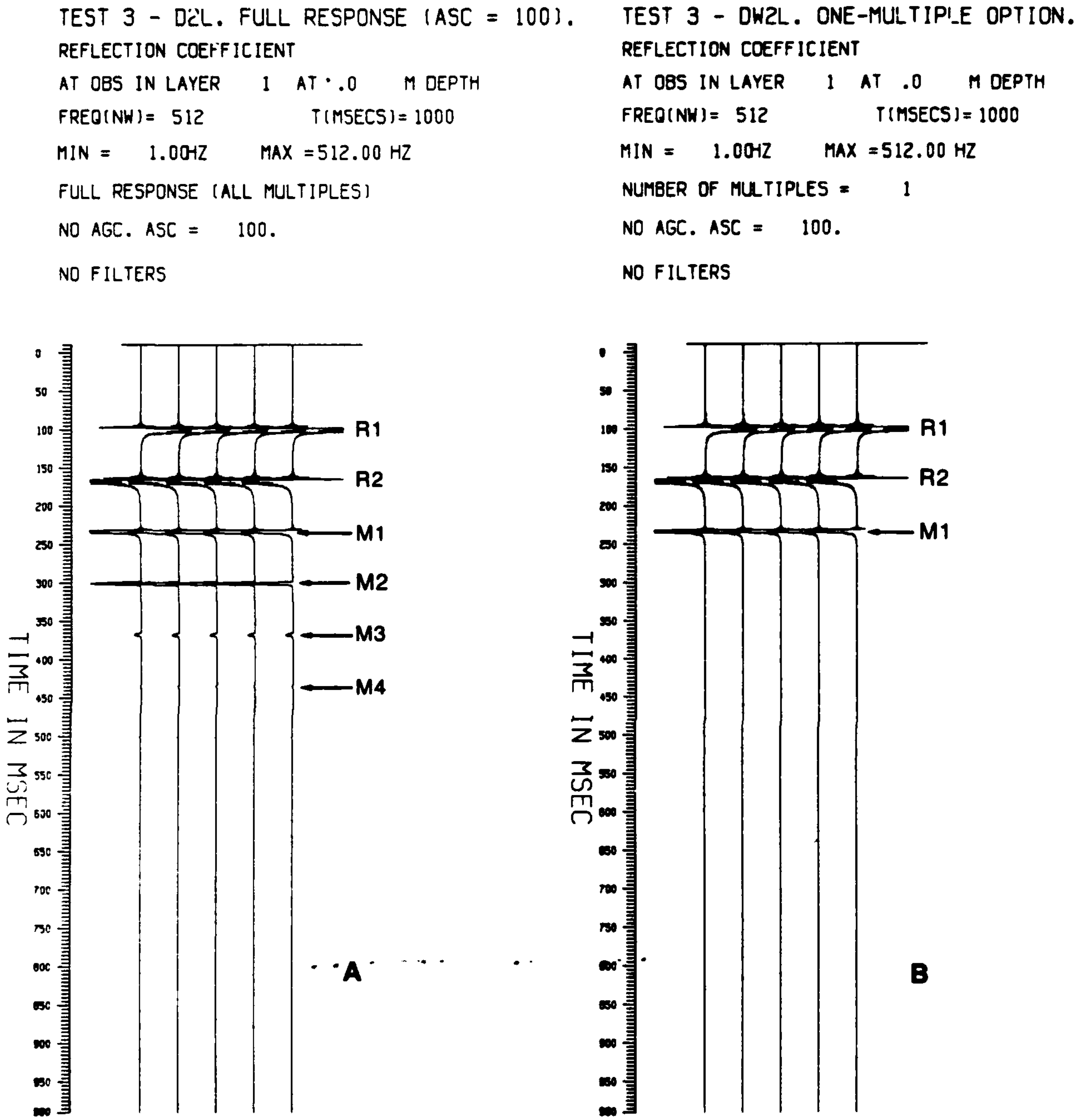
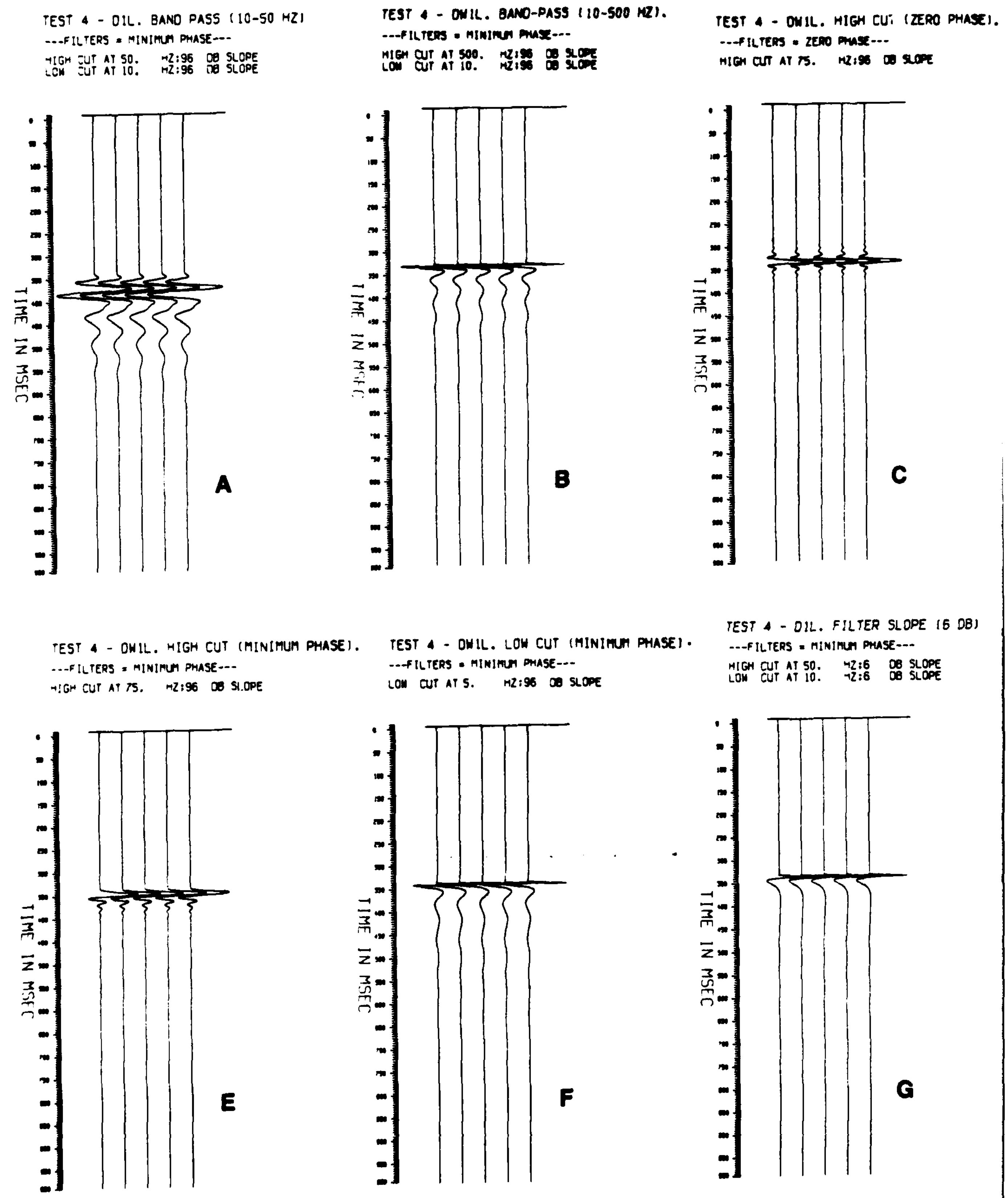


Figure 5. (Test 3) Reflections from a two-layer model (fig. 3B) for A, full-response option and B, one-multiple option. Reflections R and multiples M are identified in table 4.

The following records (table 4) were used as input to Program VISP to generate all multiples shown in figure 5A. Figure 5B, a test of one multiple only, has the same input records except MLTPL is changed from -1 to 1.


TEST 4 - DIL. BAND PASS (10-50 HZ)
---FILTERS = MINIMUM PHASE---
HIGH CUT AT 50. HZ:96 DB SLOPE
LOW CUT AT 10. HZ:96 DB SLOPE
TIME IN MSEC
A
TEST 4 - DWIL. BAND-PASS (10-500 HZ).
---FILTERS = MINIMUM PHASE---
HIGH CUT AT 500. HZ:96 DB SLOPE
LOW CUT AT 10. HZ:96 DB SLOPE
TIME IN MSEC
B
TEST 4 - DWIL. HIGH CUT (ZERO PHASE).
---FILTERS = ZERO PHASE---
HIGH CUT AT 75. HZ:96 DB SLOPE
TIME IN MSEC
C
TEST 4 - DWIL. HIGH CUT (MINIMUM PHASE).
---FILTERS = MINIMUM PHASE---
HIGH CUT AT 75. HZ:96 DB SLOPE
TIME IN MSEC
E
TEST 4 - DWIL. LOW CUT (MINIMUM PHASE).
---FILTERS = MINIMUM PHASE---
LOW CUT AT 5. HZ:96 DB SLOPE
TIME IN MSEC
F
TEST 4 - DIL. FILTER SLOPE (6 DB)
---FILTERS = MINIMUM PHASE---
HIGH CUT AT 50. HZ:6 DB SLOPE
LOW CUT AT 10. HZ:6 DB SLOPE
TIME IN MSEC
G

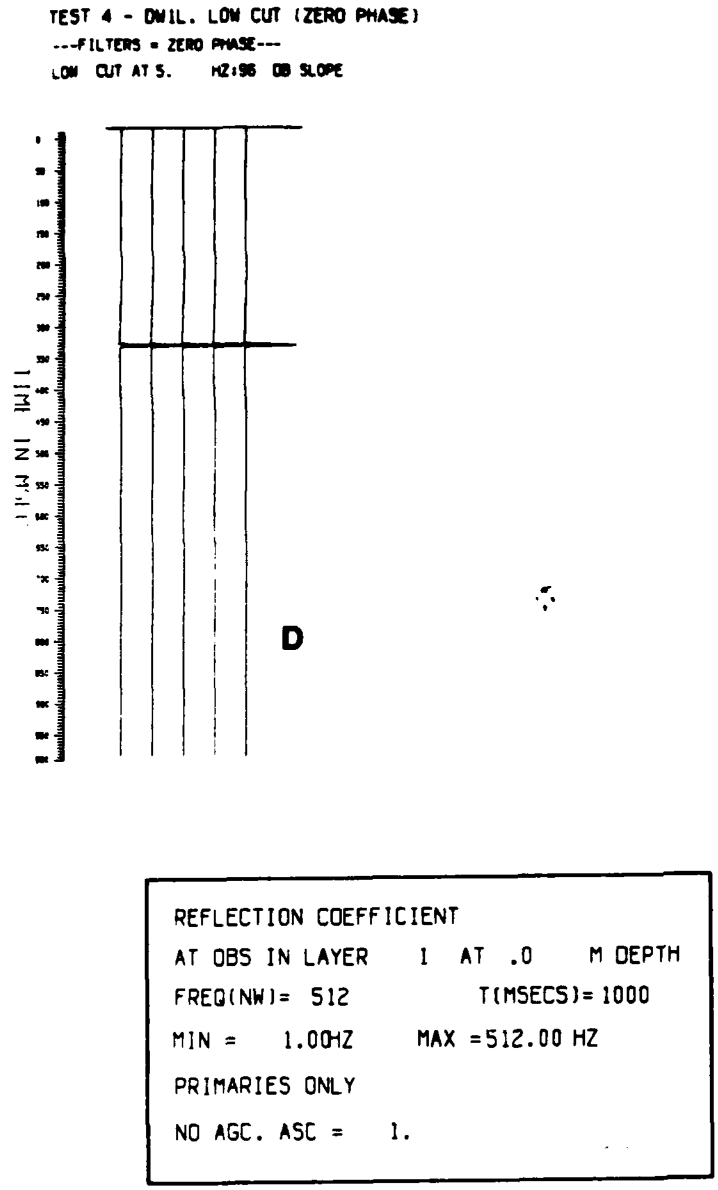


Figure 6. (Test 4) Reflections from a one-layer model (fig. 3A) used to display filter options. A, band pass of 10-50 hz; B, band pass of 10-500 hz; C, D, E, F, high and low cut with minimum and zero phase; G, band pass with 6-db slope versus 96-db slope used in A.

Test 4. Filters

A single layer 250 m thick is overlain by a half space with the same velocity (1.5 km/sec) and underlain by a half space with a velocity of 3.0 km/sec (fig. 3A). This one-layer model was used to test all filter options. Layer and half spaces have identical Q (200) and density (1.1). The single reflector permits a study of the waveform as a variety of filter parameters are changed (figs. 6A-6G).

Band Pass. Comparison of a minimum-phase band pass of 10-50 hz (fig. 6A) versus 10-500 hz (fig. 6B) shows high frequencies in the wider band pass (in the form of a spike). The dominant period on the 10-50 hz record is 30 msec (or 33 hz), about the center of the band pass. Band-pass slope was unchanged in figures 6A and 6B.

Table 5 is a copy of the input records used to generate a 10-50 hz band-pass filter (fig. 6A). To generate a 10-500 hz band-pass filter (fig. 6B), one needs only to change the cutoff frequency from 50 to 500.

Notch Filter. If a notch filter is desired, the input is similar to the band-pass option except that the "filter-type" is designated as 3 for both input lines. (See table 5.)

High and Low Cut (zero phase). A high-cut filter (low pass) set at 75 hz with 96-db slope and zero phase shows the reflection with a dominant period of 15 msec or 66 hz (fig. 6C).

A low-cut filter set at 5 hz with 96-db slope and zero phase shows the reflection characterized by a sharp spike (fig. 6D). Because the low-cut filter essentially passes all frequencies, we expect to see a spike. A zero-phase filter is noncausal; arrivals occur before the expected arrival times of 333 msec (two-way distance of 500 m at 1500 m/sec). Note that the dominant arrival (maximum amplitude) occurs at 333 msec.

Table 6 is a copy of the input records used to generate a high-cut filter (fig. 6C). To generate a low-cut filter (fig. 6D), change "filter-type" from 1 to 2 and change cutoff frequency from 75 to 5.

Table 5. Input data used to test a 10-50 hz band-pass filter (fig. 6A)

```
TEST 4 - D1L.   BAND-PASS (10-50 HZ).
EPS      SIGMA     YSIDE     XSIDE
0.001     0.1       40.        6.
CL          QL      RHO       THICKNESS
1.5        200.     1.1           0.
1.5        200.     1.1         250.
3.0        200.     1.1           0.
9999      9999     9999         9999
#TRC #FREQ T(MSEC)  IN/MSEC AMP-SCALE MLTPL AGC WINDOW
 5     512    1000.      .006        1.0      0     0      0
DETECTOR TYPE     DETECTOR LOCATION
 REFLECTION            LAYER 1
FILTER PHASE
1 (MINIMUM PHASE)
FILTER-TYPE  DB SLOPE  CUT-OFF FREQUENCY
1                96                 50
2                96                 10
```

Table 6. Input data used to test a high-cult filter with zero phase (fig. 6C)

```
TEST 4 - D1L.   HIGH CUT (ZERO PHASE).
EPS      SIGMA     YSIDE     XSIDE
0.001     0.1       40.        6.
CL          QL      RHO       THICKNESS
1.5        200.     1.1           0.
1.5        200.     1.1         250.
3.0        200.     1.1           0.
9999      9999     9999         9999
#TRC #FREQ T(MSEC)  IN/MSEC AMP-SCALE MLTPL AGC WINDOW
 5     512    1000.      .006        1.0      0     0      0
DETECTOR TYPE     DETECTOR LOCATION
 REFLECTION            LAYER 1
FILTER PHASE
0 (ZERO PHASE)
FILTER-TYPE  DB SLOPE  CUT-OFF FREQUENCY
1                96                 75
```

High and Low Cut (minimum phase). The model used to study the zero-phase filter (above) was rerun with minimum-phase option. A high-cut filter set at 75 hz with 96-db slope and minimum phase (fig. 6E) displays a slightly different waveform than the zero-phase filter. Now instead of the amplitude maximum at 333 msec, the first motion occurs at that time.

A low-cut filter set at 5 hz with minimum phase (fig. 6F) also shows a sharp peak at 333 msec. This is consistent with the minimum-phase characteristics of high frequencies occurring at the front of the waveform. Note that the minimum-phase filter passes a tail of lower frequencies (fig. 6F), although the same zero-phase filter does not.

Table 6, used to generate the high-cut filter with zero phase, may also be used to generate the minimum-phase filter. The user only needs to change "filter-phase" from 1 to 0 to obtain figure 6E. To obtain the low-cut

minimum-phase seismogram of figure 6F, the user needs to also change the filter type from 1 to 2 and the cutoff frequency from 75 to 5.

Filter Attenuation. The user may select the filter-attenuation factor (slope in decibel per octave) for high- and low-cut filters. Note the slope in decibels in a Butterworth filter is related to the number of poles (= DB/6 + 1). Variations in slope were tested on a minimum-phase 10-50 hz band-pass filter (fig. 6A). With a slope of 96 DB/octave, the 10-50 hz band pass is a boxcar-type filter yielding a reflection signal ringing at 33 hz, about the center frequency of the band pass. But the same filter with a slope of 6 DB/octave yields a reflection with a narrow spike (fig. 6G). The low slope passes a large range of frequencies to generate the spike. Figure 6G is generated with the input records shown in table 5 except for a change of slope from 96 to 6 db.

Test 5. Amplitude Scale

The user has available a simple amplitude multiplier (ASC). A two-layer model (fig. 3B), 75 m and 100 m thick, was used to test this option. For this model, reflections and multiples repeat at a 67-msec interval. For a full-response option, an ASC = 100 displays six events (fig. 5A); for an ASC = 1, only three events are readily visible (fig. 7). Note that amplitudes greater than 0.5 inch are clipped. Table 4 with amplitude scale changed from 100 to 1 will generate figure 7.

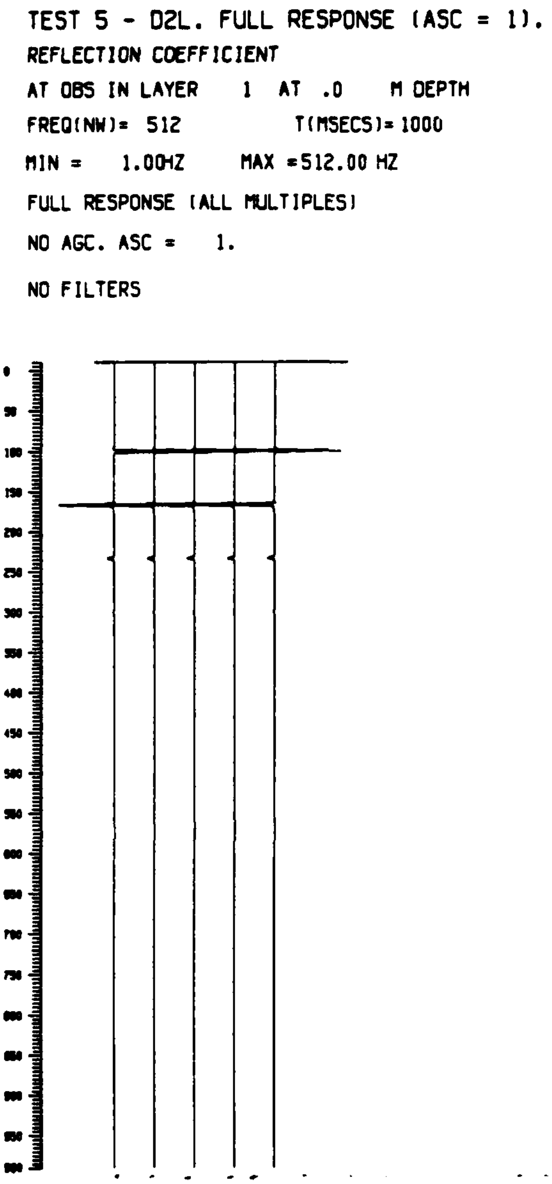


Figure 7. (Test 5) Reflections from a two-layer model (fig. 3B) with amplitude scale (ASC) = 1. Compare with amplitude scale of 100 in figure 5A.

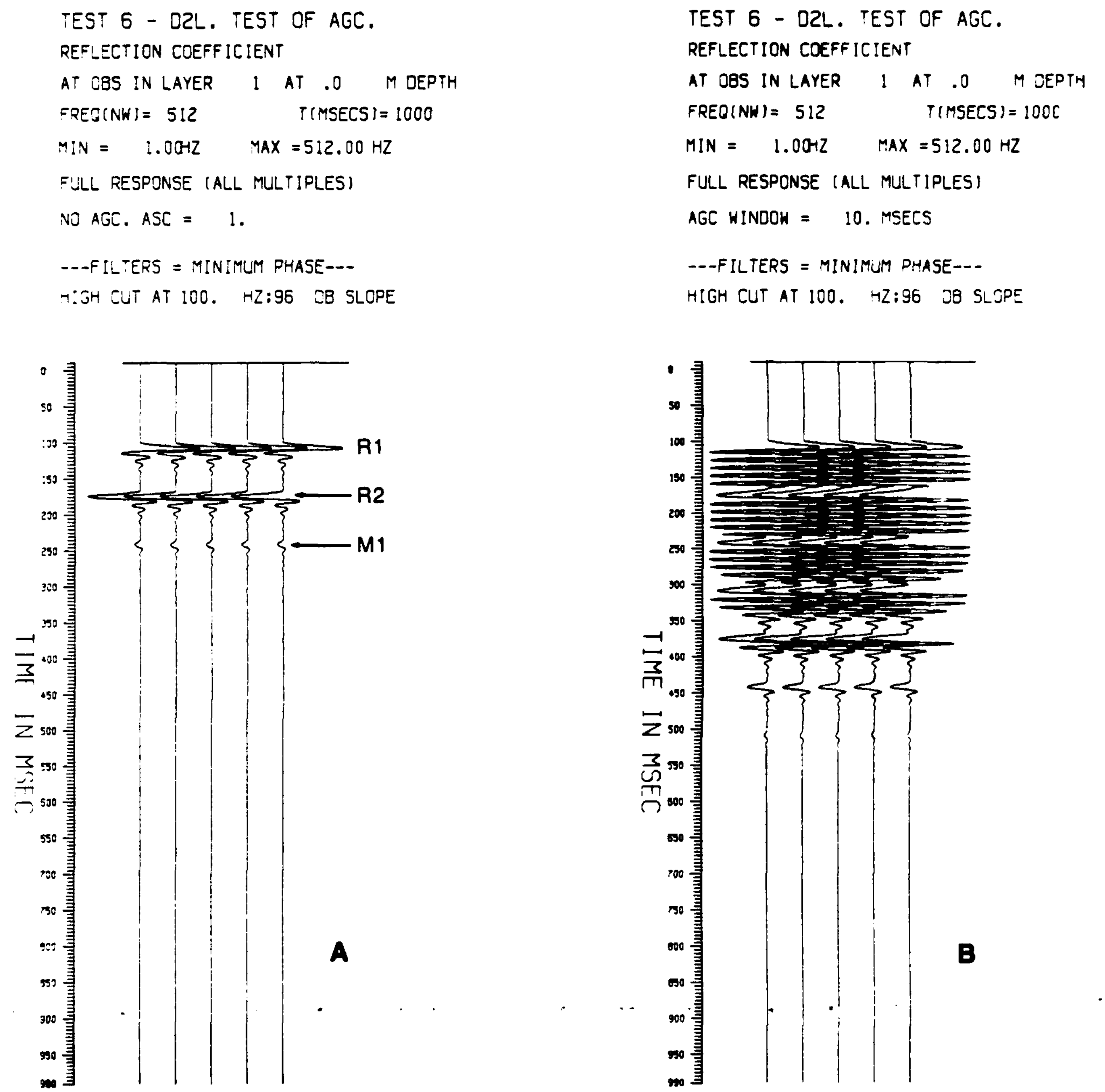
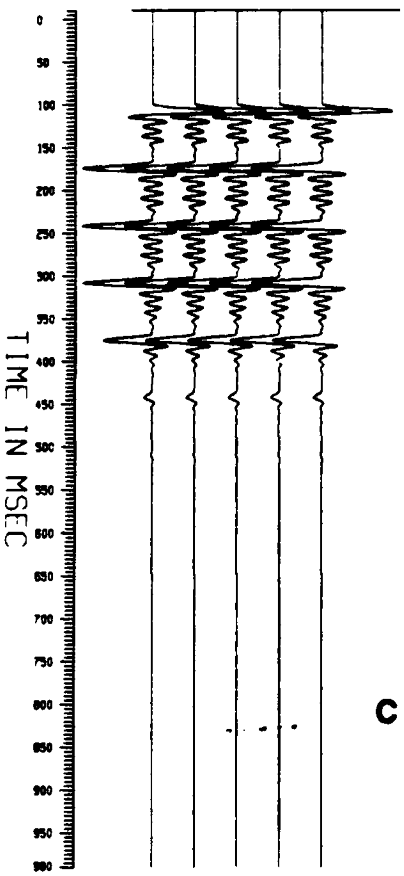


Figure 8. (Test 6) Full response from a two-layer model (fig. 3B) for A, no AGC; B, AGC with window = 10 msec; C, AGC with window = 60 msec; D, AGC with window = 500 msec. Reflections R1 and R2 and multiple M1 are identified in table 4.

TEST 6 - D2L. TEST OF AGC.
REFLECTION COEFFICIENT
AT OBS IN LAYER 1 AT .0 M DEPTH
FREQ(NW)= 512 T(MSECS)= 1000
MIN = 1.00HZ MAX =512.00 HZ
FULL RESPONSE (ALL MULTIPLES)
AGC WINDOW = 60. MSECS

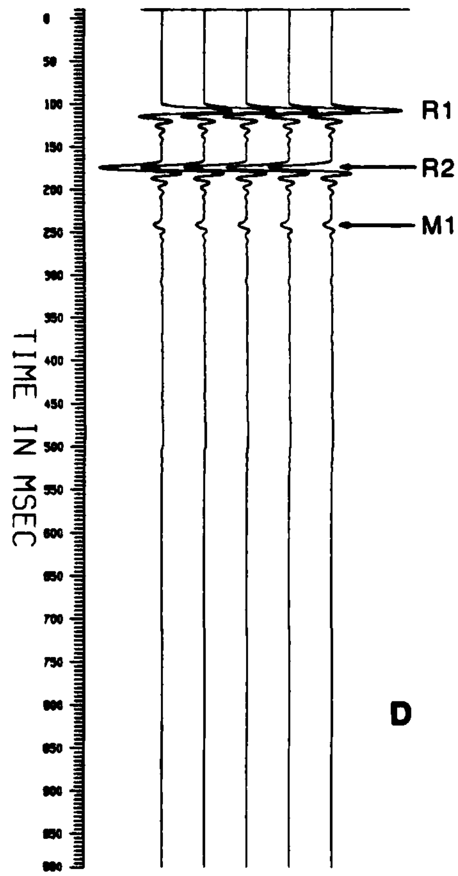
---FILTERS = MINIMUM PHASE---
HIGH CUT AT 100. HZ:96 DB SLOPE



C

TEST 6 - D2L. TEST OF AGC.
REFLECTION COEFFICIENT
AT OBS IN LAYER 1 AT .0 M DEPTH
FREQ(NW)= 512 T(MSECS)= 1000
MIN = 1.00HZ MAX =512.00 HZ
FULL RESPONSE (ALL MULTIPLES)
AGC WINDOW = 500.MSECS

---FILTERS = MINIMUM PHASE---
HIGH CUT AT 100. HZ:96 DB SLOPE



D

Figure 8 — Continued

Test 6. Automatic Gain Control

Theory. Normally we perform automatic gain control (AGC) by dividing our time series by its own RMS value computed by using a suitable window of fixed length. Therefore if $f(t)$ is the input signal, $f_c(t)$ is the AGC signal, and $f_r(t)$ is the RMS signal, we have

$$f_r(t) = \left(\int f^2(\tau)\ W(t-\tau)d\tau\right)^{1/2} \tag{10}$$

and

$$f_c(t) = f(t)/f_r(t) \tag{11}$$

where

$$W(t) = (1+\mathrm{COS}(2\pi|t|/TW) / 2TW \tag{12}$$

is a window of length TW designated by the user. For convenience, Program VISP carries out part of this computation in the frequency domain. Let $F(f^2)$ denote the Fourier transform of $f^2(t)$ and $F(W)$ the Fourier transform of $W(t)$. F^{-1} is the inverse Fourier transform. Then

$$f_c(t) = f(t) / \left\{F^{-1}[F(f^2)F(W)]\right\}^{1/2} \tag{13}$$

is the algorithm used.

Window Size. A two-layer model (fig. 3B), 75 and 100 m thick, was used to test the action of the AGC window size. Without AGC and ASC = 1, two reflectors (R1 and R2) and one multiple (M1) were observable (fig. 8A). The time interval between events is 67 msec. A small window (10 msec) yields a noisy (ringing) record that does not allow easy identification of the early events (fig. 8B). A window comparable to the expected time intervals (60 msec) still has some (ringing) noise (fig. 8C). If the window (520 msec) is one-half or more of the time span of the entire record (1000 msec), a seismogram is obtained similar to the original record without AGC (fig. 8D). The ENCODE/DECODE statements in subroutine AGCFREQ were included to circumvent difficulties with significant figures. Users may have to adjust this part of the code for their computers.

Table 7 is a copy of the input records used to create figure 8C.

Table 7. Input data used to test automatic gain control (fig. 8C)

```
TEST 6 - D2L.  TEST OF AGC.
EPS     SIGMA    YSIDE    XSIDE
0.001    0.1      40.      6.0
CL      QL       RHO       THICKNESS
1.50    200.     1.1              0.
1.50    200.     1.1             75.
3.00    200.     1.1            100.
1.50    200.     1.1              0.
9999   9999     9999          9999
#TRC #FREQ  T(MSEC)   IN/MSEC AMP-SCALE MLTPL AGC WINDOW
 5     512   1000.      .006       1.0     -1     1   60.
DETECTOR TYPE     DETECTOR LOCATION
 REFLECTION            LAYER 1
FILTER PHASE
1  (MINIMUM PHASE)
FILTER-TYPE  DB SLOPE  CUT-OFF FREQUENCY
1                96              100
```

Test 7. Computational Frequency (NW)

A model with a single layer (fig. 3A) was used to test the importance of the user's choice of computational frequencies (NW) for a fixed record length of T = 1000 msec. Figures 9A, 9B, and 9C show signal quality for computational frequencies of 16, 64, and 256 hz. A 10-50 hz minimum-phase band-pass filter was applied to the output. Comparison of NW = 16, 64, and 256 yielded seismograms that were subjectively evaluated, respectively, as: not acceptable, poor, and excellent. Although not shown here, record quality for NW = 128 was good. For NW = 512 the record was identical to 256. On a CDC Cyber 170/855 the times, excluding compilation and plotting, varied from 0.088 sec for 16 hz, 0.196 sec for 64 hz, and 0.621 sec for 256 hz.

Table 8 is a copy of the input records used to create figure 9.

Table 8. Input data used to test computational frequencies (fig. 9C)

```
TEST 7 - D1L.  ROLE OF FREQUENCIES (NW).
EPS     SIGMA    YSIDE    XSIDE
0.001    0.1      40.      6.0
CL        QL     RHO    THICKNESS
1.5      200.    1.1           0.
1.5      200.    1.1         250.
3.0      200.    1.1           0.
9999    9999    9999      9999
#TRC #FREQ T(MSEC)   IN/MSEC AMP-SCALE MLTPL AGC WINDOW
 5    256   1000.     .006       1.0     0     0     0
DETECTOR TYPE    DETECTOR LOCATION
 REFLECTION           LAYER 1
FILTER PHASE
1 (MINIMUM PHASE)
FILTER-TYPE  DB SLOPE  CUT-OFF FREQUENCY
1                48              50
2                48              10
```

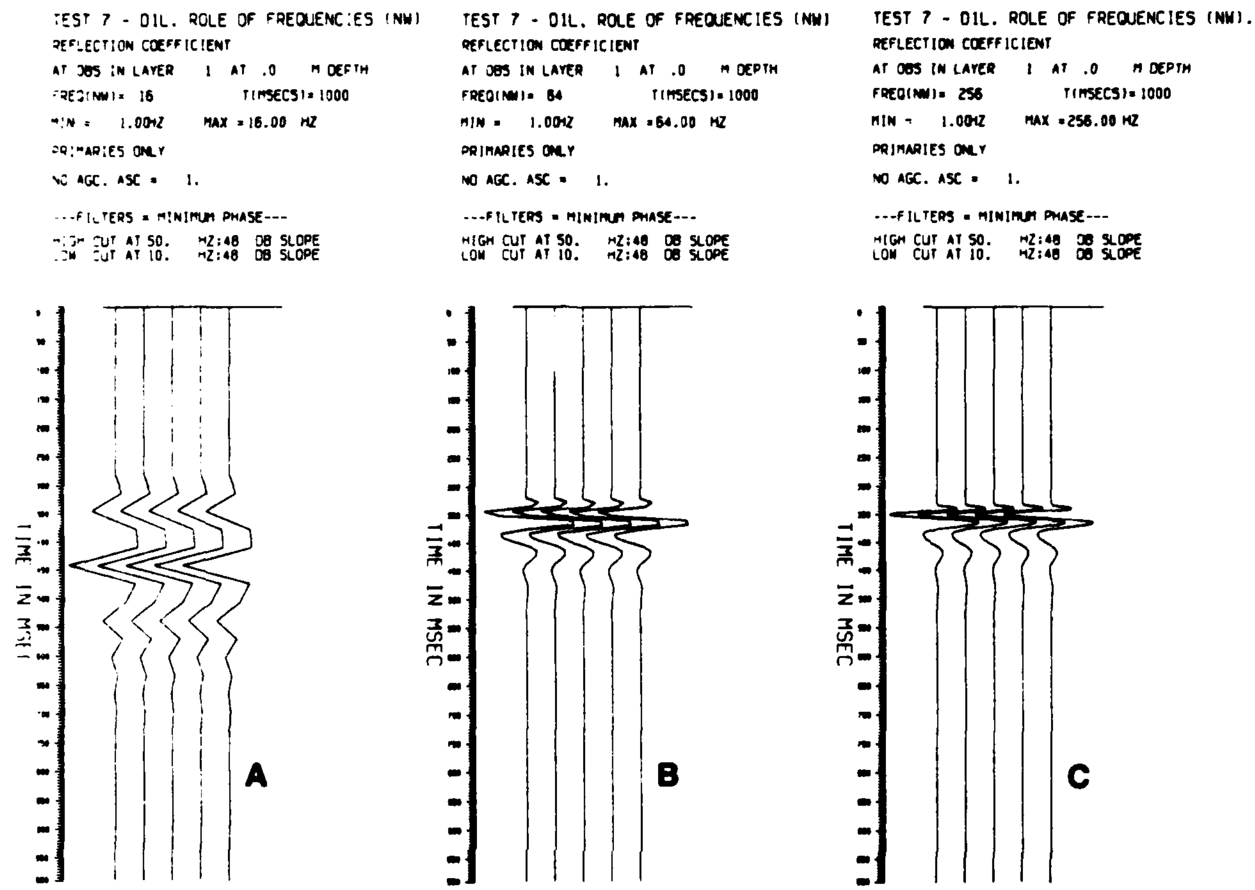


Figure 9. (Test 7) Variations of computational frequency (NW) for a single reflector model (fig. 3A) for A, NW = 16; B, NW = 64; C, NW = 256.

Test 8. Record Length (TSEC)

The interrelationship between the time span of the seismogram (TSEC) and computational frequency (NW) is examined with a six-layer model (fig. 3C). The record length (TSEC) was varied from 500 to 4000 msec in the presence of a fixed computational frequency NW = 512 hz. The minimum frequency is given by 1/TSEC, and the maximum (Nyquist) frequency is given by NW/TSEC. The results shown in figure 10 demonstrate that the user's selection of time (TSEC) must influence the appearance of the seismogram. For example, reflector R3 of the six-layer model (fig. 3C) is at a depth of 300 m and has a predicted arrival time of 550 msec. (See table 10.) The choice of record length TSEC = 500 msec is too short, and R3 arrives as a wraparound event at 50 msec (fig. 10A). Therefore all the arrivals on figure 10A other than R0, R1, and R2 are wraparound events.

A record length of 2000 msec (fig. 10B) is sufficiently long that all events (R0 to R6) are observable at their proper arrival times. Excessive time length, however, reduces the frequency of content of the signal and may not be desirable. Note that a record length of TSEC = 2000 msec (fig. 10B) gives an f_{max} of only 256 hz versus 1024 hz for the seismogram with TSEC = 500 msec. One should first choose TSEC to avoid wraparound and then choose frequency (NW) to obtain the desired (maximum) frequency content.

Table 9 is a copy of the input records used to generate figure 10B.

TEST 8 - D6L. TEST OF TSEC (= 500)
REFLECTION COEFFICIENT
AT OBS IN LAYER 1 AT .0 M DEPTH
FREQ(NW)= 512 T(MSECS)=500
MIN = 2.00HZ MAX =1024.00HZ
PRIMARIES ONLY
NO AGC. ASC = 1.
NO FILTERS

TEST 8 - D6L. TEST OF TSEC (= 2000)
REFLECTION COEFFICIENT
AT OBS IN LAYER 1 AT .0 M DEPTH
FREQ(NW)= 512 T(MSECS)=2000
MIN = .50 HZ MAX =256.00 HZ
PRIMARIES ONLY
NO AGC. ASC = 1.
NO FILTERS

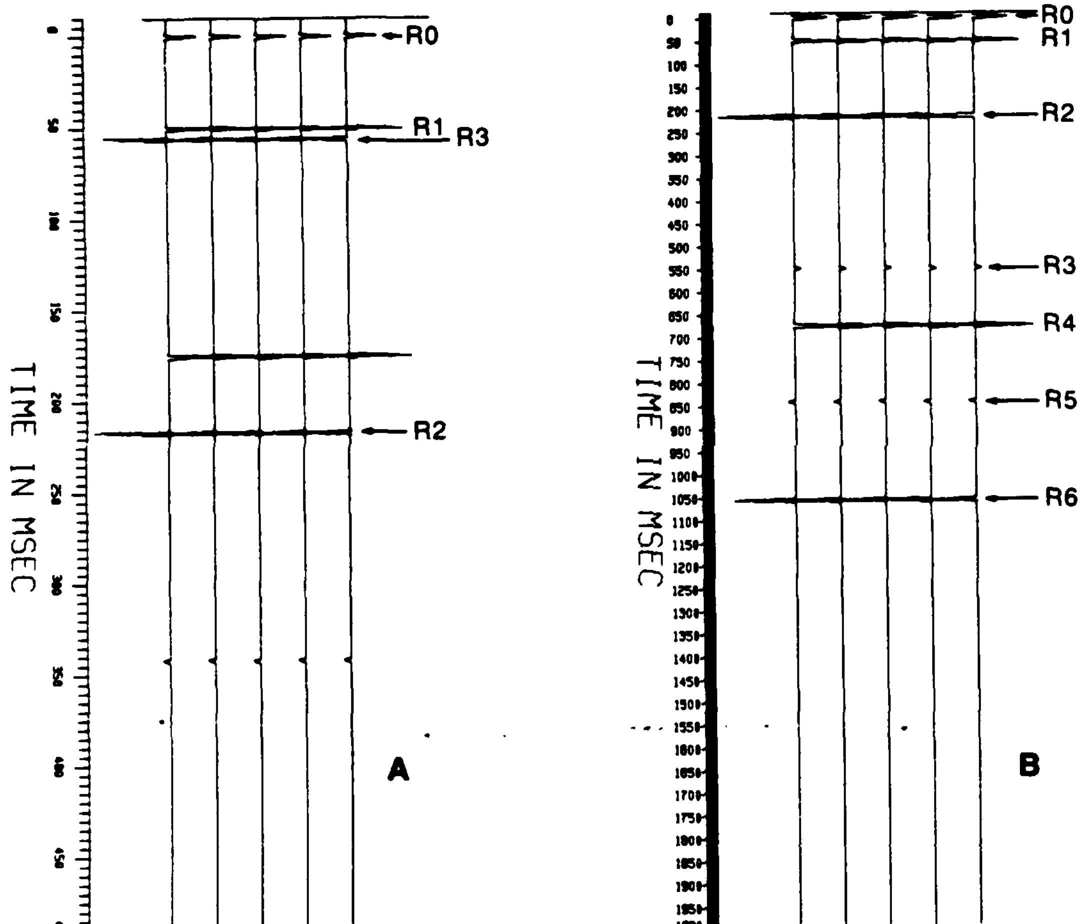


Figure 10. (Test 8) Variation of record length (TSEC) for a six-layer model (fig. 3C). A, TSEC = 500 and B, TSEC = 2000. Reflections R0 through R6 are identified in table 10.

Table 9. Input data used to test record length (fig. 10A)

```
TEST 8 - D6L.  TEST OF TSEC (= 2000)
EPS     SIGMA    YSIDE    XSIDE
0.001    0.1      40.      6.0
CL       QL      RHO     THICKNESS
1.5    10000.    1.1         0.
2.0    10000.    1.1        50.
3.0    10000.    1.1       250.
1.5    10000.    1.1       250.
1.6    10000.    1.1       100.
3.0    10000.    1.1       250.
2.8    10000.    1.1       300.
1.5    10000.    1.1         0.
9999    9999    9999      9999
#TRC #FREQ T(MSEC)  IN/MSEC AMP-SCALE MLTPL AGC WINDOW
 5    512    2000.    .003       1.0      0     0      0
DETECTOR TYPE     DETECTOR LOCATION
 REFLECTION           LAYER 1
FILTER PHASE
2 (NO FILTERS)
FILTER-TYPE  DB SLOPE  CUT-OFF FREQUENCY
```

Test 9. Arrival Times and Amplitudes

A six-layer model with constant Q = 10,000 and ρ = 1.1 (fig. 3C) was used to examine arrival times and amplitudes for the reflection-coefficient seismogram (fig. 11A). To generate a noise-free record, no multiples were included. A record length of T = 1200 msec avoided wraparound problems. To avoid amplitude distortion, the seismograms were not filtered.

Arrival times for the seven reflectors (R0 to R6) were computed from the velocities and thicknesses of the layers; R0 is from the top of layer 1, and R6 is from the top of the half space. Theoretical (predicted) amplitudes were also hand calculated for the model by using the usual relationships for reflection coefficients and two-way transmission coefficients. Study of the results (table 10) shows that the reflections arrived with correct amplitudes, polarities, and times.

Table 11 is a copy of the input records used to generate figure 11A.

Table 10. Observed and predicted times and amplitudes of reflection coefficients for a six-layer model (fig. 11A)

Reflector index	Predicted times (msec)	Observed times (msec)	Predicted amplitudes (cm) (See fig. 11A)	Observed amplitudes (cm) (normalized)
R0	0	0	+.14	.14
R1	50	50	+.20	.22
R2	217	215	-.31	-.32
R3	550	550	+.03	+.03
R4	675	675	+.25	+.27
R5	842	840	-.03	-.03
R6	1056	1050	-.22	-.25

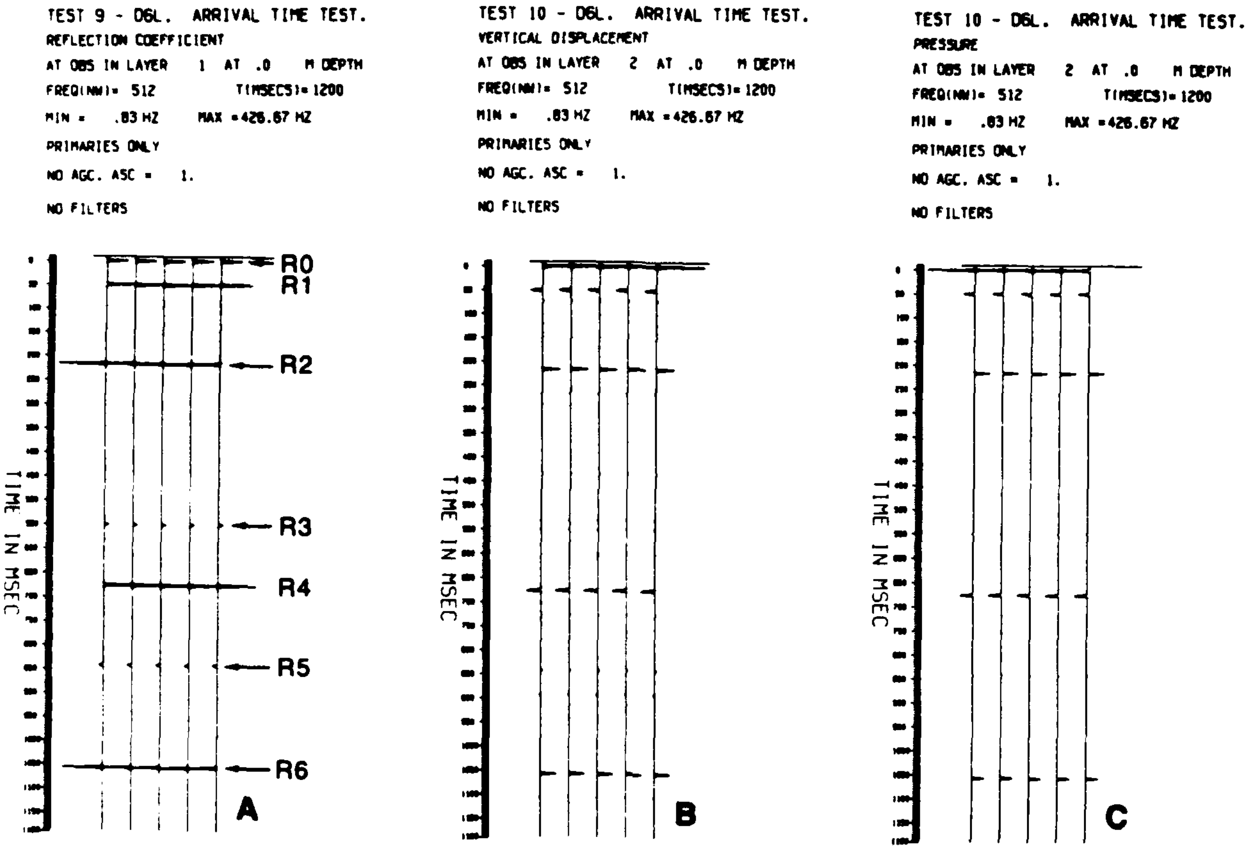


Figure 11. (Tests 9 and 10) Arrival times and amplitudes for a six-layer model (fig. 3C) for A, reflection coefficient; B, vertical displacement; C, pressure.

Table 11. Input data used to test times, amplitudes, and polarities of reflections (fig. 11A)

```
TEST 9 -  D6L.    ARRIVAL TIME TEST.
EPS       SIGMA     YSIDE     XSIDE
0.001      0.1       40.       6.0
CL         QL       RHO      THICKNESS
1.5     10000.      1.1          0.
2.0     10000.      1.1         50.
3.0     10000.      1.1        250.
1.5     10000.      1.1        250.
1.6     10000.      1.1        100.
3.0     10000.      1.1        250.
2.8     10000.      1.1        300.
1.5     10000.      1.1          0.
9999     9999     9999        9999
#TRC #FREQ T(MSEC)   IN/MSEC AMP-SCALE MLTPL AGC WINDOW
 5     512   1200.      .003       1.0      0     0   0
DETECTOR TYPE      DETECTOR LOCATION
 REFLECTION            LAYER 1
FILTER PHASE
2 (NO FILTERS)
FILTER-TYPE   DB SLOPE   CUT-OFF FREQUENCY
```

Test 10. Pressure and Vertical Displacement

The same six-layer model (fig. 3C) used to test arrival times and amplitudes for the reflection-coefficient seismogram (fig. 11A) was also used to test the waveforms and polarity of the options to plot vertical-displacement and pressure seismograms (figs. 11B and 11C). The algorithm for Program VISP is so designed that the vertical-displacement and pressure output may be activated only if the receiver is specified to be within one of the layers, not within the half space. The reflection-coefficient plot is obtained only when the receiver is at the bottom of the upper half space. Therefore, for computational purposes, the receivers for the three seismograms in figure 11 are located as follows: the reflection-coefficient output (fig. 11A) is from the receiver at the bottom of the half space (layer 1), and vertical displacement (fig. 11B) and pressure (fig. 11C) are from the top of layer 2 (in effect at the same depth as the reflection-coefficient output but across the interface).

Comparison of the three outputs shows that vertical-displacement (fig. 11B) and pressure seismograms (fig. 11C) have the same polarity for reflectors R1-R6 but differ slightly in amplitude. Reflections R1-R6 on the reflection-coefficient seismogram (fig. 10A) have the opposite polarity. Because of the receiver location, noted above, the first arrival (R0) on the reflection-coefficient seismogram is a reflection, but the first arrival on the other two seismograms is a transmitted wave of opposite polarity.

Table 11 is a copy of the input records used to generate figure 11A. To generate figures 11B and 11C one needs only to change REFLECTION to either VERTICAL or PRESSURE and LAYER 1 to LAYER 2.

Test 11. Receiver Depth

The user may place the receiver within any of the layers. The source is always within the half space, and zero time begins when the pulse encounters the top of layer 1. The six-layer model (fig. 3C), without multiples, was used to test this option. The receiver was placed at the top of layer 3 (fig. 12A), and vertical displacement was plotted. The reflection-coefficient option is not available within the layers but only at the bottom of the upper half space. Table 12 shows the computed and observed times for the primary reflections for vertical displacement (fig. 12B).

The seismogram (fig. 12B) shows the major reflectors arriving at the computed times with the appropriate polarities. Table 13 is a copy of the input records used to generate figure 12.

Table 12. Arrival times for a buried receiver (fig. 12)

Reflections	Computed times (msec)	Observed times (msec)
Direct	108	110
R3	442	440
R4	567	560
R5	733	730
R6	947	950

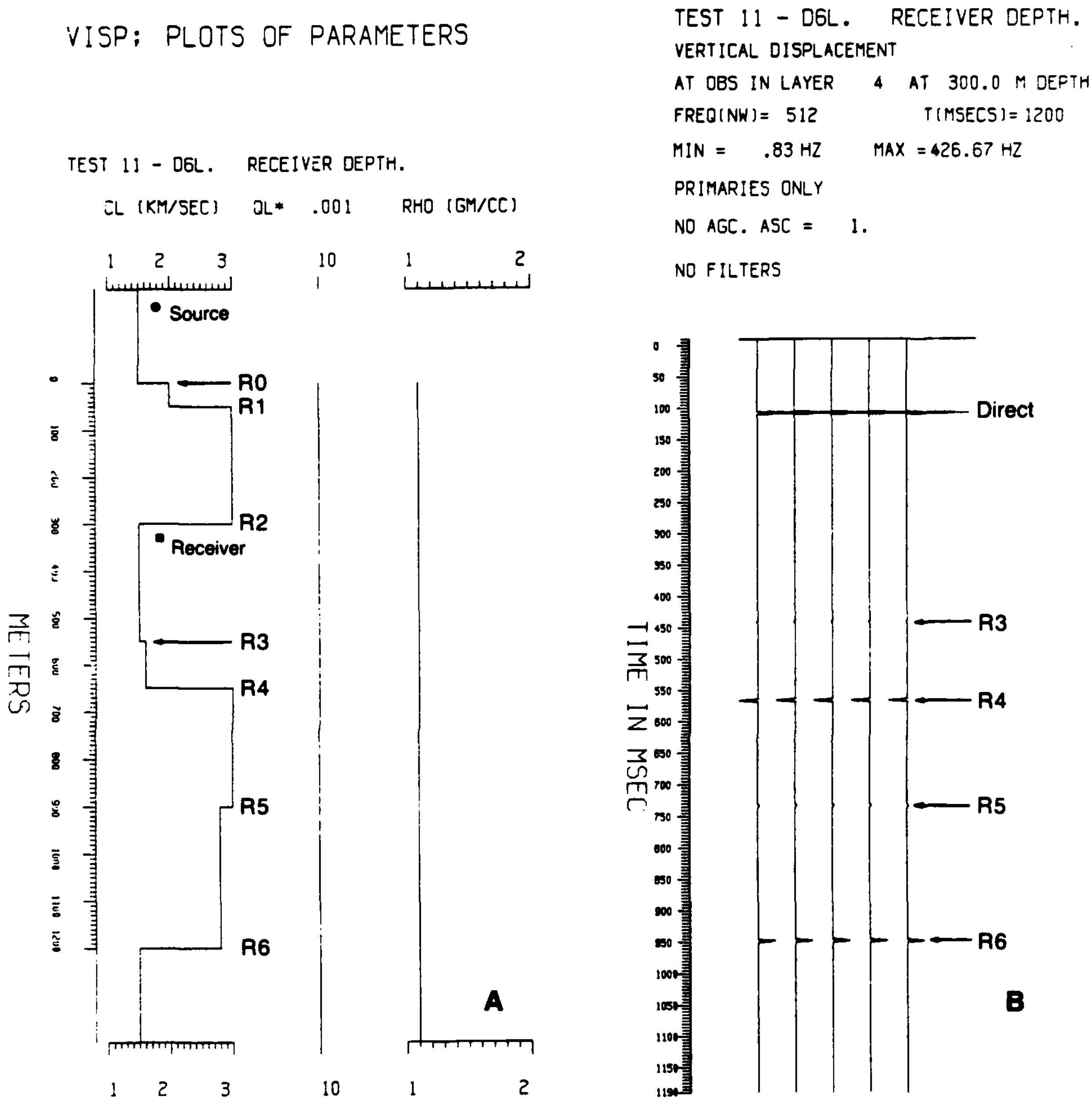


Figure 12. (Test 11) Primary reflections for a six-layer model (fig. 3C) for a receiver at the top of layer 4 (where the upper half space is termed layer 1).

Table 13. Input data used to test receiver depth (fig. 12)

```
TEST 11 - D6L.     RECEIVER DEPTH.
EPS      SIGMA      YSIDE    XSIDE
0.001     0.1        40.      6.0
CL         QL       RH0    THICKNESS
1.5      10000.     1.1         0.
2.0      10000.     1.1        50.
3.0      10000.     1.1       250.
1.5      10000.     1.1       250.
1.6      10000.     1.1       100.
3.0      10000.     1.1       250.
2.8      10000.     1.1       300.
1.5      10000.     1.1         0.
9999     9999      9999       9999
#TRC #FREQ T(MSEC)  IN/MSEC AMP-SCALE MLTPL AGC WINDOW
 5    512   1200.     .005      1.0      0    0    0
DETECTOR TYPE      DETECTOR LOCATION
 VERTICAL              LAYER  4
FILTER PHASE
2 (NO FILTERS)
FILTER-TYPE  DB SLOPE  CUT-OFF FREQUENCY
```

Test 12. Interpolated Layers

It is possible to have layer velocities uniformly increase (or decrease) within *specified depth intervals without* hand calculating the values (fig. 13). The user enters a sequence of values within a sequence of specified layers: O, N, T where O is the flag, N is the number of layers to be inserted, and T is the total thickness of the layers. Table 14 is a copy of the data used to generate figure 13.

Table 14. Input data used to interpolate layers (fig. 13)

```
TEST 12 -            INTERPOLATED LAYERS.
EPS      SIGMA      YSIDE     XSIDE
0.001     0.1        40.       6.0
CL        QL        RHO     THICKNESS
1.5      200.       1.1         0.
1.5      200.       1.1        75.
0.0        5        500         1
3.0      200.       1.1       100.
1.5      200.       1.1         0.
9999     9999      9999       9999
#TRC #FREQ T(MSEC)    IN/MSEC AMP-SCALE   MLTPL   AGC   WINDOW
 5    512   1000.       .006      1.0        0      0      0
DETECTOR TYPE     DETECTOR LOCATION
 REFLECTION           LAYER 1
FILTER PHASE
1 (MINIMUM PHASE)
FILTER-TYPE  DB SLOPE  CUT-OFF FREQUENCY
1                96           100
```

VISP: PLOTS OF PARAMETERS

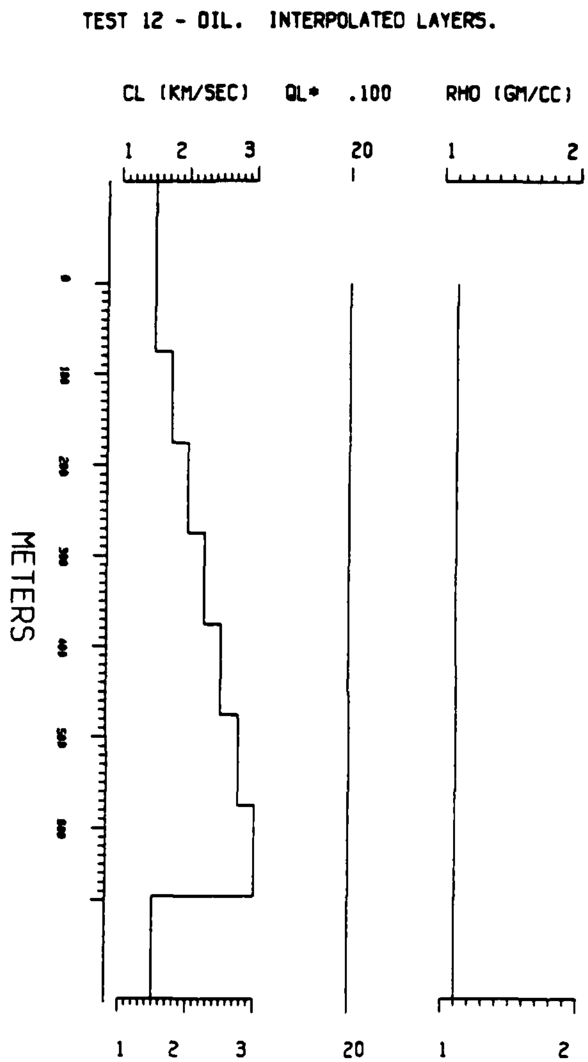


Figure 13. (Test 12) Generation of interpolated layers with uniformly increasing velocities.

Test 13. Absolute Depth

The user may specify the absolute depth below the bottom of the upper half space instead of specifying the interval thickness of each layer. The absolute depths are entered as negative values. Table 15 is a copy of the input that in effect generates the six-layer model (fig. 3C) used throughout this paper.

Table 15. Input data used to generate a six-layer model using absolute depths (fig. 3C)

```
TEST 13 - 1          ABSOLUTE DEPTH INPUT.
EPS     SIGMA        YSIDE        XSIDE
0.001    0.1          40.          6.0
CL        QL         RHO       THICKNESS
1.5    10000.        1.1              0.
2.0    10000.        1.1            -50.
3.0    10000.        1.1           -300.
1.5    10000.        1.1           -550.
1.6    10000.        1.1           -650.
3.0    10000.        1.1           -900.
2.8    10000.        1.1          -1200
1.5    10000.        1.1              0.
9999    9999        9999          9999
#TRC #FREQ T(MSEC)    IN/MSEC  AMP-SCALE   MLTPL   AGC   WINDOW
 5     512   1200.      .005        1.0        0      0      0
DETECTOR TYPE     DETECTOR LOCATION
 VERTICAL             LAYER 4
FILTER PHASE
2 (NO FILTERS)
FILTER-TYPE  DB SLOPE  CUT-OFF FREQUENCY
```

INDIANA GEOLOGICAL SURVEY GEOPHYSICAL COMPUTER PROGRAMS
ERRATA

Geophysical Computer Program 1 (Occasional Paper 10)

Page 9, 19 lines from the bottom of the page:

Second line of R(M,N,4) now reads 1+P(I+1,J+1)+P(I+1,J-1)+P(I-1,J+1)+P(I-1,J-1))/8.0

Second line of R(M,N,4) should read 1+P(I+1,J+2)+P(I+1,J-2)+P(I-1,J+2)+P(I-1,J-2))/8.0

Page 9, 4 lines from the bottom of the page:

Second line of R(M,N,11) now reads 1P(I-20,J-15)+P(I-15,J-15)+P(I+20,J+15)+P(I+15,J+20)

Second line of R(M,N,11) should read 1P(I-20,J-15)+P(I-15,J-20)+P(I+20,J+15)+P(I+15,J+20)

Page 14, line 6, which reads C(6,12)=-0.04007, may be deleted.

Geophysical Computer Program 2 (Occasional Paper 13)

Page 11, line 18:

Now reads: (1,170)ITYPE,Z(I),XI(I)

Should read: (2,230)ITYPE,Z(I),XI(I)

Page 12, after line 18:

Insert: 230 FORMAT (I1,F4.0,F4.1)

Geophysical Computer Program 3 (Occasional Paper 14)

Page 12, line 11:

Now reads: 10 A(I+MN)=A(I)

Should read: 10 A(M+K-I)=A(N+K-I)

Geophysical Computer Programs 4 and 5 (Occasional Papers 22 and 23)

Geophysical Computer Programs 4 and 5 require many significant figures. Double precision may be needed on some computers. Indiana University computers use 60-bit words.

Geophysical Computer Program 7 (Occasional Paper 29)

Subroutine MYLINE2 has been removed from the program. Delete all references to this subroutine and read all references to "11 subroutines" as "10 subroutines."

Page 38:

Now reads: 30 X 27 km region

Should read: 31 X 27 km region

Page 39:

Now reads: 6 X 40 km region

Should read: 10 X 40 km region

Page 44:

Now reads: distance 200 km

Should read: distance 20 km

Page 52:

Now reads: as a function time

Should read: as a function of time

Geophysical Computer Program 9 (Occasional Paper 40)

Page 13, line 16:

Now reads: 110 THETA=PI/2.0

Should read: 110 THETA1=PI/2.0

END

FILMED

11-85

DTIC